

PRIZE COURSE

Ecology and Ecosystems Fundamentals

A friendly, hands-on course with small Python models and practical activities.

12 Chapters | Philip Osztromok | Generated with Claude

Contents

Chapter 1	What Ecology Is	2
Chapter 2	Energy Flow	10
Chapter 3	Food Webs	17
Chapter 4	Nutrient Cycles	26
Chapter 5	Biomes of the World	35
Chapter 6	Populations	45
Chapter 7	Competition and Niches	53
Chapter 8	Predation and Cooperation	61
Chapter 9	Succession and Disturbance	69
Chapter 10	Biodiversity	77
Chapter 11	Human Impact	86
Chapter 12	Conservation and Restoration	94

Ecology and Ecosystems Fundamentals 🏆

Course 1 · Chapter 1 · What Ecology Is

Walk through any Scottish glen, London park or Norfolk reedbed and you are walking through a web of relationships: the bird eating the berry that carries the seed, the fungus feeding on the fallen branch, the rain that fills the burn that feeds the trout. **Ecology** is the science of those relationships: how living things interact with each other and with their surroundings, and so where they live, how many there are and why. This first chapter sets out the map for the whole course. You will meet the six levels at which ecologists work, learn how they count things they can never count completely, and try two real field methods in a few lines of Python. The maths is light: an average and a ratio.

> **WHAT TO TRUST IN THIS CHAPTER**
>
> The Python snippets in this chapter and in the three exercise solutions were run, and the output in the comments is what Python printed. The ecological ideas (the levels of organisation, quadrats, mark-recapture) are standard textbook material written from general knowledge, not fetched from a source for this chapter; check them against a current ecology textbook before relying on them. Every number inside the code is *invented* to show a method and is not a measurement of any real place.

Ecology Is Not the Same as Environmentalism

People use "ecology" to mean recycling, green politics or being kind to nature. Those are worthy, but they are *actions and values*. Ecology itself is a science, like physics or chemistry: it asks questions, collects evidence and tests explanations. The word comes from the Greek *oikos* (house) and *logos* (study), coined by the German biologist Ernst Haeckel in the 1860s: the study of nature's household. The science tells us how nature works; what we choose to do about it is a separate decision, which Ecology and Ecosystems Fundamentals 11 and Ecology and Ecosystems Fundamentals 12 come to at the end of the course.

Six Levels, from One Animal to the Whole Planet

Ecologists work at several scales, each one containing the one before it. Choosing the right level matters, because each asks a different kind of question.

LEVEL	WHAT IT IS	EXAMPLE	A QUESTION ASKED HERE
1. Organism	One individual living thing	One red squirrel	How does it find food and survive winter?
2. Population	All individuals of one species in one place	All red squirrels in Kielder Forest	Is the number rising or falling, and why?
3. Community	All the species living together in one place	Every plant, animal and fungus in the forest	Which species compete, eat or help each other?
4. Ecosystem	The community plus its non-living surroundings	The forest with its soil, water and climate	How do energy and nutrients move through it?
5. Biome	A large region defined by its climate and typical life	Temperate coniferous forest, tundra, desert	Why do similar climates grow similar vegetation?
6. Biosphere	All life on Earth and the places it lives	The planet's living layer	How does life change the air and oceans?

The same animal can be studied at any of them. A grey seal off the Moray coast is an organism (how deep does it dive?), part of a population (is the colony growing?), part of a marine community (what does it eat, and what eats its pups?) and part of an ecosystem (how does it move nutrients from the open sea to the coast?). A useful habit is to ask, whenever you read an ecological claim, *which level is this about?* The code below lists the levels with one example each, the sort of lookup table you will reuse in Exercise 1.

```
# The six levels of ecology, smallest to largest, with an example of each.
levels = ["organism", "population", "community", "ecosystem", "biome", "biosphere"]
examples = {
    "organism": "one red squirrel",
    "population": "all red squirrels in Kielder Forest",
    "community": "every species living in Kielder Forest",
    "ecosystem": "the forest community plus its soil, water and climate",
    "biome": "temperate coniferous forest",
    "biosphere": "all life on Earth and where it lives",
}
for i, lv in enumerate(levels, 1):
    print(f"{i}. {lv:11s} {examples[lv]}")

# Output:
# 1. organism    one red squirrel
```

```
# 2. population  all red squirrels in Kielder Forest
# 3. community   every species living in Kielder Forest
# 4. ecosystem   the forest community plus its soil, water and climate
# 5. biome       temperate coniferous forest
# 6. biosphere   all life on Earth and where it lives
```

HABITAT AND NICHE

Two more words you will meet often. A species' **habitat** is where it lives (its address: a peat bog, a rocky shore). Its **niche** is what it does and needs there (its job: what it eats, when it is active, what conditions it tolerates). Ecology and Ecosystems Fundamentals 7 is all about niches.

What Ecologists Actually Study

Whatever the level, most ecological questions boil down to a handful of themes, and these are the chapters of the rest of the course:

THEME	QUESTION	WHERE WE MEET IT
Energy	Where does the energy come from and where does it go?	Ecology and Ecosystems Fundamentals 2
Feeding links	Who eats whom, and what happens if one species vanishes?	Ecology and Ecosystems Fundamentals 3
Matter	How are carbon, nitrogen and water recycled?	Ecology and Ecosystems Fundamentals 4
Distribution	Why are different places home to different life?	Ecology and Ecosystems Fundamentals 5
Numbers	Why do populations rise, level off and crash?	Ecology and Ecosystems Fundamentals 6
Interactions	How do species compete, hunt and help one another?	Ecology and Ecosystems Fundamentals 7 and 8
Change	How do communities recover, and what do people do to them?	Ecology and Ecosystems Fundamentals 9 to 12

How Ecologists Ask Questions

Ecology is a field science, and its great difficulty is that nature is big, messy and mostly hidden. You cannot count every daisy on a hillside or every fish in the North Sea. Instead ecologists follow the usual scientific cycle: **observe** something, form a **hypothesis** (a testable explanation), **measure** carefully, and **compare** the result with what the hypothesis predicted. A good question is specific. "Why are there so few wildflowers?" is vague; "Do wildflowers grow more densely in the meadow than under the hedgerow, and is the difference bigger than chance?" can be tested with a tape measure and an afternoon.

Three habits protect ecologists from fooling themselves. They **sample** (measure a small, fair part and scale up), they **replicate** (repeat the measurement many times) and they use **controls** (compare with a similar place where the thing being tested is absent). The next two sections show the two classic sampling tools.

Counting Plants: the Quadrat

A **quadrat** is a square frame, often one metre by one metre, dropped at random positions on the ground. You count what is inside each frame, find the average, and multiply by the area of the whole site. The word "random" is doing real work: if you throw the frame only where the flowers look nicest, the answer is biased. Here are counts from ten quadrats in a 2,000 square metre field (invented numbers).

```
# Estimate the number of plants in a field from ten one-metre quadrats (invented data).
counts = [3, 0, 5, 2, 4, 1, 6, 2, 3, 4]
quadrat_area = 1.0      # square metres per frame
field_area = 2_000.0    # square metres in the whole field
n = len(counts)
mean = sum(counts) / n
density = mean / quadrat_area
estimate = density * field_area
print(f"quadrats: {n}, total plants counted: {sum(counts)}")
print(f"mean per quadrat: {mean:.2f}")
print(f"density: {density:.2f} per square metre")
print(f"estimated plants in the field: {estimate:,.0f}")

# Output:
# quadrats: 10, total plants counted: 30
```

```
# mean per quadrat: 3.00
# density: 3.00 per square metre
# estimated plants in the field: 6,000
```

Six thousand is an *estimate*, not a fact. A different set of ten frames would give a different answer. How much that matters depends on how many quadrats you use. The simulation below builds an imaginary field of 2,000 one-metre squares whose true average we know, then samples it with more and more quadrats.

```
import random
random.seed(1)                                  # fixed seed so the run is repeatable
field = [random.choice([0, 0, 1, 2, 3, 5]) for _ in range(2000)]
true_mean = sum(field) / len(field)
print(f"true mean per square metre: {true_mean:.3f}")
for n in [5, 20, 80, 320]:
    sample = random.sample(field, n)
    m = sum(sample) / n
    print(f"{n:4d} quadrats: estimate {m:.3f}  (error {abs(m-true_mean):.3f})")

# Output:
# true mean per square metre: 1.885
#    5 quadrats: estimate 2.400  (error 0.515)
#   20 quadrats: estimate 2.200  (error 0.316)
#   80 quadrats: estimate 2.138  (error 0.253)
#  320 quadrats: estimate 1.856  (error 0.028)
```

The errors shrink as the sample grows, though not perfectly smoothly: a small sample can be lucky or unlucky, so one run proves nothing on its own. That is exactly why ecologists replicate.

Counting Animals: Mark and Recapture

Animals move, hide and will not stay inside a frame, so a different trick is used. Catch a number of animals, mark them harmlessly (a dab of paint, a ring on a bird's leg, a tag on a fish), release them and let them mix back in. Later catch a second sample and see what fraction already carries a mark. If one in five of the second catch is marked, the marked animals must be about one fifth of the whole population. That gives the **Lincoln-Petersen estimate**:

Population ≈ (number marked × size of second catch) ÷ number recaptured

```
# Lincoln-Petersen estimate (invented numbers).
marked = 40         # caught, marked and released the first time
second = 50         # size of the second catch
recaptured = 8      # of the second catch, how many carried a mark
est = marked * second / recaptured
print(f"Lincoln-Petersen estimate: {est:.0f}")
for r in [4, 8, 16]:
    print(f"recaptured {r:2d}: estimate {marked * second / r:6.0f}")

# Output:
# Lincoln-Petersen estimate: 250
# recaptured  4: estimate    500
# recaptured  8: estimate    250
# recaptured 16: estimate    125
```

Notice how sensitive the answer is to the number recaptured: halve it and the estimate doubles. The method rests on assumptions, and each is a way it can go wrong.

METHOD	BEST FOR	KEY ASSUMPTION	HOW IT CAN MISLEAD
Quadrat	Plants and animals that stay put (daisies, limpets, barnacles)	Frames are placed at random	Choosing "typical" spots biases the count; too few frames give a noisy answer
Mark-recapture	Animals that move (voles, fish, butterflies)	Marks last, animals mix, population is closed	Trap-shy or trap-happy animals, lost tags, animals moving in or out

Common mistakes

Do not confuse ecology (the science) with environmentalism (a set of values and actions). Do not treat a population, a community and an ecosystem as the same thing: a population is one species, a community is all the species, and an ecosystem adds the non-living surroundings. Do not take one sample as the truth; an estimate always has uncertainty. And never place quadrats "where it looks representative": pick positions at random.

Challenges

EXERCISE 1

Make a list of eight (description, level) pairs covering all six levels, for example one oak tree, all the oaks in a wood, and the wood with its soil and rainfall. Write a short Python loop that prints each description with its level number and name. Include one pair for a marine animal.

View solution

EXERCISE 2

Two habitats each have eight one-metre quadrat counts (invented): a meadow with 12, 9, 15, 11, 8, 14, 10, 13 and a hedgerow with 4, 7, 2, 5, 6, 3, 5, 4. Each covers 400 square metres. Compute the mean, the standard error (the standard deviation divided by the square root of n), and the estimated total for each, with a rough plus-or-minus of two standard errors. Do the ranges overlap?

View solution

EXERCISE 3

Estimate the population in three invented mark-recapture studies (marked, second catch, recaptured): A 30, 40, 6; B 100, 80, 10; C 25, 30, 15. Also compute Chapman's version, (marked + 1) × (second + 1) ÷ (recaptured + 1) − 1, which is less biased when recaptures are few. Which study do you trust most, and name two assumptions that could break.

View solution

QUICK REFERENCE

- **Ecology:** the science of how living things interact with each other and their surroundings; not the same as environmentalism
- **Six levels:** organism, population, community, ecosystem, biome, biosphere
- **Habitat and niche:** where a species lives, and what it does and needs there
- **Sampling:** measure a fair, random part, replicate, and scale up; estimates carry uncertainty
- **Quadrat:** mean count per frame × (site area ÷ frame area)
- **Lincoln-Petersen:** marked × second catch ÷ recaptured

COMING NEXT

Ecology and Ecosystems Fundamentals 2 follows the energy: how sunlight becomes sugar, how it passes along food chains, and why top predators are so rare.

Ecology and Ecosystems Fundamentals

Course 1 · Chapter 2 · Energy Flow

Stand on a Scottish hillside in July and almost everything you can see is running on sunlight: the grass, the heather, the sheep eating them, the buzzard circling above. Ecosystems are not powered by the soil or by the animals. They are powered by a steady stream of energy from the Sun, and what happens to that energy as it passes from one living thing to the next decides how many plants, how many herbivores and how many predators a place can hold. This chapter follows the energy. By the end you will know why top predators are rare, why a food chain rarely has more than four or five links, and how to model the whole thing in a few lines of Python. The maths is light: multiplying by a percentage, again and again.

WHAT TO TRUST IN THIS CHAPTER

The Python snippets in this chapter and in the three exercise solutions were run, and the output in the comments is what Python printed. The ecological figures (about 1% of sunlight captured, a 10% rule of thumb, the contrast between warm- and cold-blooded animals) are textbook rules of thumb written from general knowledge, not fetched from a source for this chapter; check them against a current ecology textbook before relying on them. Every number inside the code is *assumed* or *illustrative*: it shows a method and is not a measurement of any real place.

Sunlight to Sugar: the Producers

Almost all the energy in ecosystems enters through **photosynthesis**. Plants, algae and some bacteria use sunlight to turn carbon dioxide and water into sugars, storing a little of the light energy as chemical energy. Organisms that do this are called **producers** (or autotrophs, "self-feeders"). On land the producers are plants, from Highland grasses to rainforest trees. In the sea they are mostly microscopic **phytoplankton**, which is why the North Sea, rich in plankton, supports so much life.

It surprises most people that plants capture only a small slice of the light that falls on them: usually around one per cent or less over a year, because most light is reflected, passes through, warms the leaf or is the wrong colour. Ecologists keep two accounts of what plants do capture.

TERM	STANDS FOR	MEANING
GPP	Gross primary production	All the energy the producers fix by photosynthesis
Respiration	R	The share the producers burn to stay alive
NPP	Net primary production	GPP minus respiration: the energy left over to build new leaves, wood and roots, and so the food available to everything else

NPP is the figure that matters to the rest of the food chain, because consumers can only eat what the plants have built. Roughly speaking, NPP is high where it is warm, wet and sunny (rainforests, estuaries, reefs) and low in deserts, in the cold polar regions and in the open ocean far from land, where nutrients are scarce. Ecology and Ecosystems Fundamentals 5 returns to this on the tour of the biomes, and Ecology and Ecosystems Fundamentals 4 shows how the nutrient cycles hold the ceiling on production.

Who Eats Whom: Trophic Levels

A **food chain** is a line showing who eats whom. Each step is a **trophic level** (from the Greek *trophe*, nourishment). Counting from the producers upward:

LEVEL	NAME	EATS	NORTH SEA EXAMPLE	HIGHLAND EXAMPLE
1	Producer	Sunlight (makes its own food)	Phytoplankton	Heather, grass
2	Primary consumer (herbivore)	Producers	Zooplankton	Red deer, mountain hare
3	Secondary consumer (carnivore)	Herbivores	Herring	Wildcat, stoat
4	Tertiary consumer (top carnivore)	Other carnivores	Gannet, seal	Golden eagle

Two groups sit outside the neat ladder. **Decomposers** (fungi and bacteria) feed on dead material from every level and release its nutrients back to the soil; without them the cycle would stop. **Detritivores** such as earthworms and woodlice eat dead plant and animal remains directly.

In many ecosystems, including most woodlands, more energy flows through the dead-matter route than through the grazing route. Real animals also rarely stick to one level: a fox that eats a vole and a blackberry is a consumer at two levels. Ecology and Ecosystems Fundamentals 3, on food webs, deals with that tangle; here we keep to simple chains.

> **ENERGY FLOWS, NUTRIENTS CYCLE**
>
> A point worth remembering all course: energy goes one way. It arrives as sunlight, passes along the chain and leaves as heat, and the heat cannot be reused by plants. Matter such as carbon and nitrogen, on the other hand, is recycled again and again. The first idea is this chapter; the second is Ecology and Ecosystems Fundamentals 4.

Where the Energy Goes: Trophic Efficiency

When a deer eats grass, only part of the grass's stored energy ends up as deer. Some of the plant is never eaten (roots, tough stems). Some of what is eaten passes straight through and leaves as dung, which feeds decomposers. Of the part absorbed, most is burnt in respiration to run the deer's body and keep it warm, and only the remainder goes into growth and offspring. Only that last portion is available to the wolf, or in Scotland these days, the stalker, the eagle or the fox that eats a carcase.

The fraction of energy passed from one trophic level to the next is the **trophic efficiency**. A famous rule of thumb, often credited to the American ecologist Raymond Lindeman's work on a Minnesota lake in the early 1940s, says it is about **10%**: each level gets roughly a tenth of the one below. Measured values vary widely, from a few per cent to about 20%, so the 10% rule is a handy round number and not a law of nature.

```
# Energy passing up a North Sea food chain when each step keeps 10% (illustrative numbers
sunlight = 1_000_000.0   # kJ of sunlight landing on one square metre in a year (assumed
capture = 0.01           # about 1% captured by photosynthesis (illustrative)
efficiency = 0.10        # the "10% rule" of thumb

levels = ["Producers (phytoplankton)", "Primary (zooplankton)", "Secondary (herring)", "
energy = sunlight * capture
for name in levels:
    print(f"{name:26s} {energy:10.1f} kJ")
    energy *= efficiency
```

```
# Output:
# Producers (phytoplankton)     10000.0 kJ
# Primary (zooplankton)          1000.0 kJ
# Secondary (herring)             100.0 kJ
# Tertiary (gannet)                10.0 kJ
```

Read the output: from 10,000 kJ at the base, the gannet level has 10. That is one-thousandth. The same arithmetic explains why there are far fewer eagles than deer and far more grass than either.

The Energy Pyramid, and Why Chains Are Short

Draw each trophic level as a bar whose width is the energy it holds and stack them, and you get a **pyramid of energy**. An energy pyramid is always a true pyramid, wide at the bottom and

occasionally be upside down: a single oak can feed thousands of insects, and in the ocean the standing stock of tiny plankton can be smaller than the animals feeding on it, because plankton reproduce so quickly. Energy per year is the fair comparison.)

Because each link loses most of the energy, there is a limit to how many links a chain can have. The sketch below asks how many levels survive if the top level needs at least 1 kJ per square metre per year to exist at all (an invented threshold).

```
# How many trophic levels can a given efficiency support?
# Rule for this sketch: a level needs at least 1 kJ per square metre per year to exist.
npp = 10_000.0   # kJ per square metre per year at the base (illustrative)
for eff in [0.05, 0.10, 0.20]:
    energy = npp
    levels = 1
    while energy * eff >= 1.0:
        energy *= eff
        levels += 1
    print(f"efficiency {eff:4.0%}: {levels} trophic levels, top level gets {energy:9.2f}

# Output:
# efficiency   5%: 4 trophic levels, top level gets      1.25 kJ
# efficiency  10%: 5 trophic levels, top level gets      1.00 kJ
```

```
# efficiency  20%: 6 trophic levels, top level gets      3.20 kJ
```

A small change in efficiency adds or removes a whole level, which fits what we see: chains of three to five links are normal, and long marine chains tend to be found where producers are very productive and the animals are cold-blooded. The take-away is that top predators are rare because there is so little energy left for them, not because they are hard to find. It also means that people eating *lower* on the chain can be fed from far less land, which is the ecology behind the claim that a plant-based diet needs less farmland than a meat-heavy one.

Why Some Animals Waste Less: Warm Blood Versus Cold

Efficiency is not the same everywhere. A mammal or a bird keeps its body at a constant high temperature, and that burns a great deal of its food. A lizard, a fish or an insect lets its body temperature follow its surroundings and spends far less. The result is a big difference in what is left over for growth, shown here for two animals given the same meal (all figures illustrative).

```
# Compare two animals eating the same 100 kJ of food.
# Figures are textbook-style illustrations, not measurements of real species.
animals = {
    "insect (cold-blooded)": 0.40,   # share of assimilated energy turned into growth
    "small mammal (warm-blooded)": 0.02,
}
intake = 100.0
assimilation = 0.50   # share of eaten food actually absorbed (assumed)
for name, prod in animals.items():
    absorbed = intake * assimilation
    growth = absorbed * prod
    print(f"{name:28s} absorbed {absorbed:5.1f} kJ, new body tissue {growth:5.2f} kJ ({g

# Output:
# insect (cold-blooded)        absorbed  50.0 kJ, new body tissue 20.00 kJ (20.0% of int
# small mammal (warm-blooded)  absorbed  50.0 kJ, new body tissue  1.00 kJ ( 1.0% of int
```

	COLD-BLOODED (INSECT, FISH)	WARM-BLOODED (BIRD, MAMMAL)
Body heat	Follows the surroundings	Held constant by burning food
Share of energy into growth	High (tens of per cent)	Very low (a few per cent)
As prey	Passes on much more energy	Passes on little
Pays off when	Food is scarce or temperatures are mild	Being active in the cold matters, as in a Scottish winter

Common mistakes

Do not treat the 10% rule as exact: real efficiencies vary by ecosystem and species. Do not say energy is "recycled" (it is not; matter is). Do not forget the decomposers: dung and dead bodies are energy too, passed to a different branch of the web rather than lost. And an energy pyramid is not the same as a pyramid of numbers: one tree can host thousands of insects.

Challenges

EXERCISE 1

A five-link chain starts with 20,000 kJ per square metre per year at the producers (assumed). Print the energy at every level, and its share of the base, for trophic efficiencies of 5%, 10% and 15%, then say how much more energy the top level receives at 15% than at 5%.

View solution

EXERCISE 2

An invented pond has these yearly energy figures in kJ per square metre: algae 8,000, water fleas 1,200, minnows 150, perch 30, heron 1.5. Work out the trophic efficiency of every link in a loop, find the leakiest one, and suggest one biological reason why a link to a warm-blooded bird might be low.

View solution

EXERCISE 3

A predator needs 20,000 kJ per year of new tissue (assumed). For efficiencies of 5%, 10% and 20% and chains of 2, 3 and 4 links above the producers, calculate the producer energy required and the area of land needed if each square metre fixes 5,000 kJ a year (assumed). What does the table say about why top predators need such large territories?

View solution

QUICK REFERENCE

- **Producers:** capture sunlight by photosynthesis; usually about 1% of incoming light or less
- **GPP, NPP:** NPP = GPP minus the producers' own respiration; it is the food supply for everything else
- **Trophic levels:** producers, primary, secondary and tertiary consumers; decomposers work on the dead from every level
- **Trophic efficiency:** the share passed up a level; about 10% as a rule of thumb, with a wide real range
- **Energy pyramid:** always narrows upward; this is why top predators are few and chains are short
- **Energy flows one way; matter cycles**

COMING NEXT

Ecology and Ecosystems Fundamentals 3 is about food webs: real feeding networks, keystone species and trophic cascades.

Ecology and Ecosystems Fundamentals

Course 1 · Chapter 3 · Food Webs

Chapter 2 followed energy up a single, tidy chain: grass, deer, eagle. Real life is messier. The eagle also eats hares and the occasional fox, the fox eats hares and carrion, and the hare eats heather as well as grass. Draw every one of those arrows and the chain becomes a net. This chapter is about that net, the **food web**: how to describe it, how to measure how tangled it is, and why a few species hold the whole structure up far out of proportion to their numbers. We will build small webs in Python, remove species to see what happens, and watch a famous domino effect called a trophic cascade. The maths stays light: counting and simple percentages.

> **WHAT TO TRUST IN THIS CHAPTER**
>
> The Python snippets here and in the three exercise solutions were run, and the output in the comments is what Python printed. The webs and the kelp model are *invented* to show a method: the species are real but the diets are simplified and every rate is illustrative. The ecological history (the starfish experiments, the sea otters, wolves and beavers) is written from general knowledge, not fetched from a source for this chapter, and some of it, especially the wolves, is still debated by ecologists. Check it against a current textbook before relying on it.

From Chains to Webs

A **food web** is the set of all the feeding links in a community. Each species is a **node** and each "eats" relationship is a **link** (an arrow from the eaten to the eater, or, as in our code, a list of what each species eats). Compared with a chain, a web has two important features. First, most consumers have **several** food sources, so they can switch when one becomes scarce. Second, most species are eaten by **several** predators, so no single predator controls them completely.

	FOOD CHAIN (CHAPTER 2)	FOOD WEB (THIS CHAPTER)
Shape	One line of arrows	A network with many crossing arrows
Diet	One food per species	Several foods per species (generalists) or few (specialists)
If a species vanishes	Everything above it starves	Others can often switch food, but not always

	FOOD CHAIN (CHAPTER 2)	FOOD WEB (THIS CHAPTER)
Best for	Following energy loss level by level	Understanding stability and indirect effects

Here is a small invented Highland web written as a Python dictionary, with a few lines to count its species and links. This is a handy way to hold any web: each key eats the species in its list, and the producers have an empty list.

```
# A tiny food web as a dictionary: each animal maps to the things it eats (invented High
web = {
    "heather": [],
    "grass": [],
    "red deer": ["heather", "grass"],
    "mountain hare": ["heather", "grass"],
    "stoat": ["mountain hare"],
    "fox": ["mountain hare", "red deer"],
    "golden eagle": ["mountain hare", "stoat", "fox"],
}
species = len(web)
links = sum(len(prey) for prey in web.values())
print(f"{species} species, {links} feeding links")
for name, prey in web.items():
    eaten_by = [p for p, diet in web.items() if name in diet]
    print(f"{name:14s} eats {len(prey)}, eaten by {len(eaten_by)}")
connectance = links / (species * (species - 1))
print(f"connectance (links / possible links): {connectance:.3f}")

# Output:
# 7 species, 10 feeding links
# heather        eats 0, eaten by 2
# grass          eats 0, eaten by 2
# red deer       eats 2, eaten by 1
# mountain hare  eats 2, eaten by 3
# stoat          eats 1, eaten by 1
# fox            eats 2, eaten by 1
# golden eagle   eats 3, eaten by 0
# connectance (links / possible links): 0.238
```

Connectance is the fraction of all possible links that actually exist: here 10 out of 7 × 6 = 42, or 0.238. Real webs are far from fully connected; most species eat only a small fraction of what is available, which is why connectance values are usually well below 0.5. Notice also that the mountain hare is eaten by three different predators and eats two foods: it is the busy middle of this web, and a hare crash would be felt in several places at once.

Trophic Level in a Web: Fractions Appear

In a chain, trophic levels are whole numbers. In a web an omnivore feeds at several levels, so ecologists give it a fractional level: **1 plus the average level of its diet**. A fox that eats half mice (level 2) and half owls (level 3) sits at 1 + 2.5 = 3.5. The recursion is natural to code.

```
# Fractional trophic level = 1 + average level of the diet.
web = {
    "grass": [],
    "mouse": ["grass"],
    "vole": ["grass"],
    "owl": ["mouse", "vole"],
    "fox": ["mouse", "owl"],
}
levels = {}
def level(name):
    if name in levels:
        return levels[name]
    diet = web[name]
    levels[name] = 1.0 if not diet else 1 + sum(level(p) for p in diet) / len(diet)
    return levels[name]
for name in web:
    print(f"{name:6s} trophic level {level(name):.2f}")

# Output:
# grass  trophic level 1.00
# mouse  trophic level 2.00
# vole   trophic level 2.00
# owl    trophic level 3.00
# fox    trophic level 3.50
```

The fox lands at 3.50, between owl and mouse, just as the sum said. People are omnivores too, and the average human diet sits somewhere between 2 and 3 depending on how much meat it contains, which links back to the energy argument in Ecology and Ecosystems Fundamentals 2.

Not All Species Are Equal: Keystones

If you counted links alone you might guess that the best-connected species matter most. Often the opposite is true: a few species have an effect on their community far larger than their numbers or their mass would suggest. These are **keystone species**, named by analogy with the central stone of an arch: remove it and the arch falls, though it is only one stone among many. The idea comes from the American ecologist Robert Paine, who in the 1960s kept removing a starfish (the purple sea star) from rocky shore plots on the Pacific coast. Without the starfish, mussels spread over the rock, crowded out most other species and the plot lost much of its variety.

EXAMPLE	WHERE	WHAT IT DOES	IF IT IS LOST
Purple sea star	Pacific rocky shores	Eats mussels, keeping them from monopolising the rock	Mussels take over; diversity falls
Sea otter	North Pacific kelp forests	Eats sea urchins	Urchins overgraze kelp, leaving bare “urchin barrens”
Beaver	Rivers of Europe and North America	An *ecosystem engineer*: dams create ponds and wetlands	Fewer pools, less wetland habitat for fish, amphibians and birds
Wolf	Yellowstone, USA	Eats and frightens deer and elk	Grazers browse young trees heavily (the strength of this effect is debated)

Beavers deserve a Scottish note. They were hunted to extinction in Britain several centuries ago. A trial reintroduction began at Knapdale in Argyll in 2009, and a separate wild population on the River Tay in Tayside grew from escapes, and beavers in Scotland have since been given official protected status. Watching what their dams do to local wildlife is a live experiment in keystone ecology. Keystone species are not always predators: the beaver is a builder, not a hunter.

> **LINK COUNTS CANNOT FIND KEYSTONES**
>
> A species that is eaten by many and eats many may still be easy to replace. A keystone is defined by the *effect of removing it*, not by how many arrows touch it. That is why ecologists do removal

experiments, and why the next sections test removals in code.

Removing a Species: the Bottom-Up Test

A simple thought experiment is to delete one species from the web and see who is left with nothing to eat. This is a **secondary extinction** test. It catches loss of food, working from the bottom upward, and it is a rough tool, but it shows an important idea: webs with many alternative foods are more robust than webs where an animal depends on a single species.

```
# Remove one species and see who loses ALL their food (a secondary extinction).
web = {
    "kelp": [],
    "urchin": ["kelp"],
    "limpet": ["kelp"],
    "sea otter": ["urchin"],
    "fish": ["kelp", "limpet"],
    "seal": ["fish"],
}
def lose(web, removed):
    gone = {removed}
    changed = True
    while changed:
        changed = False
        for sp, diet in web.items():
            if sp in gone or not diet:
                continue
            if all(p in gone for p in diet):
                gone.add(sp)
                changed = True
    return gone - {removed}
for sp in web:
    print(f"remove {sp:10s} -> also lost: {sorted(lose(web, sp)) or 'nobody'}")

# Output:
# remove kelp       -> also lost: ['fish', 'limpet', 'sea otter', 'seal', 'urchin']
# remove urchin     -> also lost: ['sea otter']
# remove limpet     -> also lost: nobody
# remove sea otter  -> also lost: nobody
# remove fish       -> also lost: ['seal']
```

```
# remove seal       -> also lost: nobody
```

Removing the kelp, the one producer, collapses everything, which is no surprise. Removing the urchin starves the otter, which has no other food in this simplified web. Removing the limpet harms nobody, because the fish also eat kelp. The surprise is in the sea otter row: removing it loses *nobody* by this test, yet in real kelp forests the loss of otters is disastrous. The test is blind to effects that run from the top *down*, and those are the subject of the next section.

Trophic Cascades: Effects That Run Downhill

A **trophic cascade** is an indirect effect that passes down through the levels. Take away a top predator, its prey increases, and that prey then eats more of whatever lies below it. In North Pacific kelp forests the sea otter eats urchins; when otters were hunted for their fur, urchin numbers rose, grazed the kelp to the rock and left an "urchin barren" with far fewer fish and other animals. When otters returned, kelp came back. The same logic works in lakes (fish-eating fish, small fish, water fleas, algae) and is a main reason fisheries managers worry about removing top predators from the sea.

The sketch below is a yearly model with invented rates: otters reduce the urchin growth rate, and urchins graze the kelp, which also regrows a little each year. It runs eight years for four different otter populations.

```
# Trophic cascade: otters eat urchins, urchins eat kelp. Yearly steps, invented rates.
def run(otters, years=8):
    urchins, kelp = 100.0, 100.0
    for y in range(years):
        urchins *= 1.5 - 0.02 * otters      # urchin growth is cut by otter predation
        urchins = min(urchins, 1000.0)
        kelp -= 0.05 * urchins              # urchins graze kelp
        kelp += 10                          # kelp regrows a little
        kelp = max(0.0, min(kelp, 100.0))
    return urchins, kelp
for otters in [0, 10, 25, 40]:
    u, k = run(otters)
    print(f"otters {otters:3d}: urchins {u:7.1f}, kelp {k:5.1f}")

# Output:
```

```
# otters   0: urchins  1000.0, kelp   0.0
# otters  10: urchins   815.7, kelp  19.9
# otters  25: urchins   100.0, kelp 100.0
# otters  40: urchins     5.8, kelp 100.0
```

With no otters the urchins hit their cap of 1,000 and the kelp is gone. With 25 otters the urchin population simply holds steady at 100 and the kelp stays full. Notice that the otters never touch the kelp: their effect on it is entirely indirect, passed through the urchins. That is the signature of a cascade, and it is why the bottom-up test above could not see it. Real cascades are rarely this clean, since other predators, disease, storms and food supply all push on the same numbers, but the structure of the effect is the same.

Why Webs Are Fragile and Robust at Once

Ecologists have long argued about whether complexity makes ecosystems stable. The practical picture is mixed. A web with many alternative links tolerates the loss of a random species well, because most species have a back-up food. But the same web can be badly damaged by the loss of one particular species, a keystone or a base producer, and by the arrival of a new one: an invasive predator in a web whose animals have never met it can cascade through everything (Ecology and Ecosystems Fundamentals 11 returns to this). Rare species are not always the least important ones.

> Common mistakes
> Do not call a species a keystone just because it is common or well connected: the test is the size of the effect of removing it. Do not assume cascades always happen: they are strong in some ecosystems (lakes, kelp forests) and weak in others, and the Yellowstone wolf story is still argued over. Do not forget that the food web in a diagram is a simplification: real diets change with season, age and what is available. And a chain is not a web: do not draw one line of arrows and call it the ecosystem.

Challenges

EXERCISE 1

An invented loch web: algae and water plants (producers); snails eat both; water fleas eat algae; minnows eat water fleas and snails; trout eat minnows, snails and water fleas; otters eat trout and minnows; herons eat minnows and trout. Store it as a dictionary, print the number of species, the number of links and the connectance, rank every species by total links (eats plus eaten by), and list the top predators (species eaten by nobody).

View solution

EXERCISE 2

Using the same loch web, run the removal test for every species with the `lose` function from this chapter. Which removal does the most damage, and why are so few removals harmful in this test? Explain in a sentence why this test would not find a keystone predator.

View solution

EXERCISE 3

Reuse the kelp cascade model and run it for 0 to 40 otters in steps of 5. Then find the smallest whole number of otters that keeps the kelp at 50 or more after eight years, and say what the sharp change tells you about thresholds in cascades.

View solution

QUICK REFERENCE

- **Food web:** the network of all feeding links; most species have several foods and several predators
- **Connectance:** links divided by possible links (species × (species − 1)); real webs are well below 1
- **Fractional trophic level:** 1 plus the average level of the diet
- **Keystone species:** an outsized effect on the community, judged by what happens when it is removed
- **Ecosystem engineer:** a species such as the beaver that changes the habitat itself
- **Trophic cascade:** an indirect effect running down through the levels; strong in some ecosystems, weak in others

COMING NEXT

Ecology and Ecosystems Fundamentals 4 follows the matter instead of the energy: how carbon, nitrogen, phosphorus and water cycle through ecosystems.

Ecology and Ecosystems Fundamentals

Course 1 · Chapter 4 · Nutrient Cycles

Chapters 2 and 3 followed *energy* and *feeding* through a community. Energy, you will remember, only travels one way: sunlight in, heat out. Matter is different. The atoms in your body are not new; carbon that was in the air a few years ago is now in the bread you ate, and some of it will leave you as breath. Every ecosystem recycles a limited stock of carbon, nitrogen, phosphorus and water, passing it between living things, soil, rock, water and air. This chapter looks at those **nutrient cycles**: what they are made of, how fast they turn, and what happens when people push more matter into them than they can handle. The maths stays light: subtraction, division and a few percentages, all done in short Python snippets.

WHAT TO TRUST IN THIS CHAPTER

The Python snippets here and in the three exercise solutions were run, and the output in the comments is what Python printed. The numbers are *round, illustrative figures* chosen to show a method, not current measurements: real estimates of carbon stores, water volumes and nitrogen flows differ between sources and change over time, and the fertiliser and soil model uses invented rates. The ecological background is written from general knowledge, not fetched from a source for this chapter. Check any figure you want to rely on against a current textbook or a national agency before quoting it.

Energy Flows, Matter Cycles

The key contrast to keep in mind is that the Earth gets a steady supply of *energy* from the Sun but, apart from the odd meteorite, no new *matter*. So whatever the carbon, nitrogen and phosphorus in living things are, they must be reused again and again. Ecologists draw a nutrient cycle with two kinds of thing: **pools** (also called reservoirs or stores), the places where a nutrient sits, and **fluxes**, the flows that move it from one pool to another. A pool is a quantity (so many tonnes); a flux is a rate (so many tonnes a year).

	ENERGY (CHAPTER 2)	MATTER (THIS CHAPTER)
Direction	One way: Sun, then organisms, then heat	Round and round in cycles
Supply	Continuous from the Sun	A fixed stock, reused

	ENERGY (CHAPTER 2)	MATTER (THIS CHAPTER)
Lost as	Waste heat at each level	Nothing lost, only moved to another pool
Key question	How much reaches each trophic level?	How big are the pools and how fast do the fluxes run?

Two ideas make cycles easy to reason about. The first is **conservation**: atoms are moved, not made or destroyed, so for any pool, *what comes in minus what goes out equals the change in the pool*. The second is **residence time**: the size of the pool divided by the flow through it, which is the average time a molecule stays put. A small pool with a big flow turns over quickly; a huge pool with a trickle of flow holds its contents for ages.

The Carbon Cycle

Carbon is the backbone of every organic molecule. Plants take carbon dioxide from the air by **photosynthesis** and build it into sugars and wood. Animals, fungi and bacteria release it again by **respiration** and decay. Carbon also dissolves into the sea, and over very long times it is locked away in rock, coal, oil and gas. The atmosphere holds only a tiny fraction of the planet's carbon (carbon dioxide is around 0.04% of the air by volume, about 420 parts per million, or ppm), yet that small pool is the one that sets how much heat the Earth traps. Scotland has its own large carbon store: peat. Wet blanket bog, such as the Flow Country of the far north, locks up carbon in slowly accumulating peat because the waterlogged plant remains do not fully decay. Drain or burn a bog and that carbon returns to the air.

CARBON POOL	ROUGHLY HOW MUCH	HOW FAST IT TURNS OVER
Atmosphere	Under 1,000 GtC	Years (plants and oceans swap it constantly)
Plants	A few hundred GtC	Years to centuries (grass versus trees)
Soils and peat	Over 1,500 GtC	Decades to thousands of years
Oceans	Tens of thousands of GtC	Centuries to millennia
Fossil fuels and rock	Very large	Millions of years, unless burnt

(GtC means gigatonnes of carbon: a gigatonne is a billion tonnes. The table is deliberately vague on the numbers because sources disagree.) Burning fossil fuels moves carbon from the slowest

pool to the fastest one, the air, faster than the land and sea can take it back. The snippet below does the accounting using round numbers: emissions, minus what plants and soils absorb, minus what the oceans absorb, leaves a gain in the atmosphere. Because the balance rule is just subtraction, you can check the whole story in a few lines.

```
# Carbon in the air: sources, sinks and what is left over (round, illustrative numbers).
emissions = 10.0      # fossil fuels and cement, GtC a year
land_sink = 3.1       # extra carbon absorbed by plants and soils
ocean_sink = 2.5      # extra carbon dissolving into the sea
atmosphere_gain = emissions - land_sink - ocean_sink
GTC_PER_PPM = 2.13    # about 2.13 GtC raises CO2 by 1 part per million
print(f"atmosphere gains {atmosphere_gain:.1f} GtC a year")
print(f"that is about {atmosphere_gain / GTC_PER_PPM:.1f} ppm a year")
print(f"airborne fraction: {atmosphere_gain / emissions:.0%}")
ppm = 420.0
for year in range(2030, 2061, 10):
    ppm += 10 * atmosphere_gain / GTC_PER_PPM
    print(year, f"{ppm:.0f} ppm")

# Output:
# atmosphere gains 4.4 GtC a year
# that is about 2.1 ppm a year
# airborne fraction: 44%
# 2030 441 ppm
# 2040 461 ppm
# 2050 482 ppm
# 2060 503 ppm
```

So of every 10 tonnes of carbon emitted, about 4.4 stay in the air: the **airborne fraction** is roughly 44% in this simplified sum, close to the figure ecologists and climate scientists often quote. The rest is taken up by plants and soils (3.1) and the sea (2.5). That is why the carbon dioxide level rises steadily rather than leaping: the sinks are working hard, just not hard enough. A caution about the last four lines: they simply assume the sinks never change, which is the weakest assumption in the model. Warming, drought and fire can all make a sink shrink or even turn into a source.

The Water Cycle and Residence Time

Water is the simplest cycle to picture and the best one for learning residence time. The Sun's heat **evaporates** water from the oceans and, through plants, from the land (**transpiration**). It condenses into clouds, falls as rain or snow, and returns to the sea by rivers and groundwater. Almost all water, around 97%, is in the oceans, and most of the freshwater is locked in ice. Scotland, with its wet west-coast climate, is a good place to see the cycle at work: Atlantic air picks up moisture, drops it on the western hills and leaves the east much drier.

The next snippet divides each pool by its flow to get a residence time. The volumes are rounded, textbook-style figures in cubic kilometres, and the flows are the yearly amounts passing through.

```
# Residence time = size of the pool / flow through it (rounded textbook-style volumes in
pools = {
    "atmosphere": (12_900, 505_000),
    "rivers": (2_000, 45_000),
    "soil moisture": (16_500, 70_000),
    "ice sheets": (24_000_000, 2_500),
    "oceans": (1_335_000_000, 505_000),
}
for name, (volume, flow) in pools.items():
    days = volume / flow * 365
    if days < 365:
        print(f"{name:14s} {days:10.1f} days")
    else:
        print(f"{name:14s} {days / 365:10.1f} years")

# Output:
# atmosphere            9.3 days
# rivers               16.2 days
# soil moisture        86.0 days
# ice sheets         9600.0 years
# oceans             2643.6 years
```

A molecule of water spends only about nine days in the air before it rains out, but thousands of years in the oceans and ice sheets. This is why weather changes in days while sea level and ice change over centuries. The same sum works for a loch: a small loch with fast inflow flushes itself in months; a big, slow one takes years, which matters a great deal when something harmful gets

into it, as the next sections show. (If you meet the word *turnover*, it is simply the reciprocal: flow divided by pool.)

The Nitrogen Cycle

Nitrogen is a paradox. About 78% of the air is nitrogen gas (N_2), yet plants and animals mostly cannot use it, because the two atoms are held together by a very strong triple bond. Living things need nitrogen to make proteins and DNA, so the nitrogen cycle is largely a story of **converting nitrogen into usable forms** and back. Bacteria do most of the work.

STAGE	WHAT HAPPENS	WHO DOES IT
Fixation	N_2 gas becomes ammonia a plant can use	Bacteria (some live in root nodules of peas, beans and clover), and lightning
Uptake	Plants absorb ammonium and nitrate through their roots	Plants; animals get their nitrogen by eating them
Ammonification	Dead matter and waste release ammonium	Decomposer bacteria and fungi
Nitrification	Ammonium is turned into nitrate	Soil bacteria, needing oxygen
Denitrification	Nitrate is turned back into N_2 gas	Bacteria in waterlogged, low-oxygen soil

Because fixation is slow and nitrogen is often scarce, nitrogen limits plant growth in many ecosystems, which is why farmers have long used clover in rotations and manure on fields. In the early twentieth century the **Haber-Bosch process** began making ammonia fertiliser from nitrogen gas on an industrial scale. It feeds a large share of the world today, but it also roughly doubled the amount of usable nitrogen entering the land. Nitrate is very soluble, so whatever the crop does not take up can **leach** into streams and groundwater.

A one-field model shows the problem. Each year the soil receives some nitrogen from rain and natural sources plus fertiliser; the crop takes what it can, up to a limit; and a share of what is left washes away. The rates are invented, but the pattern is the point. The “crop takes” figure is the uptake in the tenth year.

```
  # A leaky nitrogen store: soil mineral nitrogen, crop uptake and leaching (kg N per hect
```

```
def run(fertiliser, years=10):
    soil = 50.0
    leached_total = 0.0
    for y in range(years):
        soil += 20 + fertiliser          # 20 from rain, fixation and soil, plus the fer
        uptake = min(soil, 180.0)        # the crop can use at most 180
        soil -= uptake
        leached = 0.4 * soil             # 40% of what is left washes out each year
        soil -= leached
        leached_total += leached
    return leached_total / years, uptake
for f in [0, 100, 160, 200, 300]:
    leach, up = run(f)
    print(f"fertiliser {f:3d} kg/ha: crop takes {up:5.1f}, leaches {leach:5.1f} kg/ha a y

# Output:
# fertiliser   0 kg/ha: crop takes  20.0, leaches   0.0 kg/ha a year
# fertiliser 100 kg/ha: crop takes 120.0, leaches   0.0 kg/ha a year
# fertiliser 160 kg/ha: crop takes 180.0, leaches   5.0 kg/ha a year
# fertiliser 200 kg/ha: crop takes 180.0, leaches  39.0 kg/ha a year
# fertiliser 300 kg/ha: crop takes 180.0, leaches 124.1 kg/ha a year
```

With no fertiliser the crop is starved (it takes only the 20 kg that the soil receives each year after

that the crop cannot use the extra, and the losses climb quickly: 39 kg a year at 200, more than 120 at 300. That excess nitrogen ends up in rivers and the sea, where it fuels the algal growth described next. A farmer who wants a good yield and clean water aims for the point where the crop is fed and the leaching curve has not yet taken off, which is what “precision” fertiliser use means in practice.

The Phosphorus Cycle: No Gas Allowed

Phosphorus is needed for DNA, cell membranes and the energy molecule ATP, but its cycle differs from carbon and nitrogen in one crucial way: **it has no significant gas phase**. Phosphorus comes from rock. Slow weathering releases phosphate into soil and water, plants take it up, animals eat the plants, and decay returns it to the soil. Over geological time some washes to the sea floor and is buried in sediment, and only uplift of new rock completes the circle. So the cycle is very slow and phosphorus is the limiting nutrient in many freshwater lakes.

That makes lakes sensitive. When sewage, detergents and farm run-off add phosphorus (and nitrogen), algae and water plants boom. The excess growth blocks the light, dies, and is decomposed by bacteria that use up the dissolved oxygen, so fish suffer. This process is called **eutrophication**. Scottish lochs, such as Loch Leven in Kinross-shire, have been studied for it, and cutting the phosphorus entering the water is the usual remedy. Recall residence time: a loch with a long residence time recovers slowly even after the source is cut, because the phosphorus already inside takes years to flush out (Exercise 1 tries this).

	CARBON	NITROGEN	PHOSPHORUS	WATER
Big store	Rock, oceans, fossil fuels	Air (N_2)	Rock and sediment	Oceans
In the air?	Yes (CO_2)	Yes, but unusable	No (dust only)	Yes (vapour)
Speed	Fast in life, slow in rock	Moderate; bacteria drive it	Very slow	Fast
Human problem	Rising CO_2	Nitrate leaching, extra N	Eutrophication	Shortages, pollution

Decomposers: the Cycle's Engine

Every one of these cycles depends on **decomposers**: fungi, bacteria and small animals such as earthworms that break dead matter back into simple nutrients. Without them dead leaves would pile up for ever and the nutrients in them would stay locked away. Decomposition is faster in warm, moist conditions, which is why a tropical forest floor is clear of leaves while a cold, waterlogged bog preserves them as peat. Decomposers also connect this chapter to the previous ones: the dead matter they feed on is a share of the energy that Chapter 2 said never reaches the next trophic level, and in many ecosystems the "detritus" web (see Ecology and Ecosystems Fundamentals 3) carries more energy than the grazing web.

Common mistakes

Do not say nutrients are "used up": they are moved between pools, and the pool that matters is the one the organism can reach. Do not mix up a pool (an amount) and a flux (an amount per time): GtC and GtC a year are different quantities. Do not assume the sinks stay the same size in a warming world. Do not call nitrogen scarce just because the air is full of it: most of it is unusable until bacteria fix it.

And do not treat fertiliser as harmless extra: the part the crop cannot take does not vanish, it moves downstream.

Challenges

EXERCISE 1

Two invented lochs: a large one holds 2.0 km³ and has 0.5 km³ of inflow a year; a small one holds 0.01 km³ and has 0.04 km³ of inflow a year. Store them in a dictionary, work out each residence time in years and in days, and say which loch would clear a phosphorus pollution problem faster once the source is stopped, and why.

View solution

EXERCISE 2

Reuse the carbon budget with the land sink at 3.1 and the ocean sink at 2.5 GtC a year. For emissions of 10, 7.5, 6 and 3 GtC a year, print the yearly atmosphere gain in GtC and ppm, and the change over ten years. Then work out what emission level would stop the rise, and say why real sinks make this answer less certain than it looks.

View solution

EXERCISE 3

Reuse the one-field nitrogen model and search fertiliser rates from 0 to 300 kg/ha in steps of 5. Find the highest rate that keeps leaching under 10 kg/ha a year, then print the leaching at that rate and the next two steps up, and explain in a sentence what the result says about applying more fertiliser than a crop needs.

View solution

QUICK REFERENCE

- **Pool:** an amount of a nutrient stored somewhere; **flux:** an amount moving per unit time

- **Conservation:** in minus out equals the change in the pool
- **Residence time:** pool size divided by flow: about 9 days for water in the air, thousands of years in the ocean
- **Carbon:** photosynthesis takes it in, respiration and burning release it; about 44% of emissions stay in the air in the chapter's round sums
- **Nitrogen:** abundant as gas but unusable until bacteria fix it; excess nitrate leaches
- **Phosphorus:** comes from rock, has no gas phase and cycles very slowly; excess causes eutrophication
- **Decomposers:** fungi and bacteria that return nutrients from dead matter to the soil

COMING NEXT

Ecology and Ecosystems Fundamentals 5 zooms out to the whole planet: how temperature and rainfall shape the great biomes of forest, grassland, desert, tundra and ocean.

Ecology and Ecosystems Fundamentals

Course 1 · Chapter 5 · Biomes of the World

Chapter 4 ended with a hint: decomposition is fast in warm, moist places and slow in cold, waterlogged ones, which is why a tropical forest floor is bare while a bog keeps its dead plants as peat. That is one small example of a bigger pattern. Climate, above all *temperature* and *rainfall*, decides what kind of ecosystem a place can support, and so how fast energy flows and nutrients cycle there. The great ecosystem types of the planet are called **biomes**. This chapter tours them: what makes a forest a forest and a desert a desert, why Scotland looks the way it does, and how the ocean fits in. The maths stays light: comparing numbers, counting months and a little multiplication, all in short Python snippets.

WHAT TO TRUST IN THIS CHAPTER

The Python snippets here and in the three exercise solutions were run, and the output in the comments is what Python printed. The climate figures for the named places are *rounded, plausible values written for practice*, not weather-station records, and the biome cut-offs and production figures are simplified, illustrative numbers. Real biome boundaries are blurry and different textbooks draw them differently. The ecological background is written from general knowledge, not fetched from a source for this chapter, so check any figure you want to rely on against a current textbook or a national agency.

What Is a Biome?

A **biome** is a large region defined by its typical climate and the kind of vegetation that grows there, regardless of which exact species live in it. Forests in Scotland, Japan and Chile are all "temperate forest" even though they share almost no species. Biomes are a useful label because similar climates push plants towards similar shapes: broad evergreen leaves in the wet tropics, small tough leaves in dry places, needles in the cold north. The shape follows the conditions: organisms are fitted to their surroundings.

Two numbers do most of the work. **Mean annual temperature** sets how long and how vigorous the growing season is, and **annual precipitation** (rain plus snow) sets how much water plants can use. Plot every land location on a graph with temperature along one axis and rainfall up the other and the biomes fall into recognisable regions; the best-known version is the **Whittaker**

diagram. Other factors matter too, including soil, fire, grazing animals and altitude, but climate comes first.

BIOME	CLIMATE	TYPICAL VEGETATION	EXAMPLE
Tundra	Very cold, low precipitation, short cool summer	Mosses, lichens, dwarf shrubs; frozen ground (permafrost)	Arctic Siberia, northern Canada, high Scottish mountain tops
Boreal forest (taiga)	Long cold winters, short mild summers	Conifers such as spruce, pine and larch	Scandinavia, Russia, Canada
Temperate forest	Mild, moderate rain all year, four seasons	Broadleaf trees such as oak, beech and birch (deciduous), plus some evergreens	Western Europe, eastern USA, Japan
Temperate grassland	Hot summers, cold winters, too dry for many trees	Grasses and herbs, grazed by large herbivores	North American prairies, Eurasian steppe
Mediterranean shrubland	Mild wet winters, hot dry summers	Tough, often aromatic evergreen shrubs; fire-adapted	Mediterranean basin, California, central Chile, Cape Town region
Savanna	Warm all year with a distinct wet and dry season	Grassland with scattered trees; fire and grazing keep it open	East and southern Africa, northern Australia
Tropical rainforest	Hot and wet all year	Tall dense evergreen forest with many layers	Amazon, Congo basin, Southeast Asia
Desert	Very little precipitation (under about 250 mm a year)	Sparse, drought-adapted plants	Sahara, Arabia, Atacama, Antarctica

Notice that a desert is defined by dryness, not heat: the Antarctic interior gets so little precipitation that it counts as a **polar desert**. Notice also that Scotland sits in a curious spot. At about 56°N, Edinburgh is on the same latitude as parts of Labrador in Canada, which is subarctic, yet Scotland has mild, wet temperate forest climate. The reason is the ocean: the warm North Atlantic Drift (the north-eastern arm of the Gulf Stream system) carries heat towards north-west Europe, and the prevailing south-westerly winds bring it ashore along with moisture.

Guessing a Biome from Two Numbers

The snippet below turns the idea into a function: give it a mean annual temperature in degrees Celsius and a rainfall total in millimetres, and it returns a biome name using a handful of cut-offs. The cut-offs are simplified and invented for teaching, so treat them as a model, not a map. The places and their figures are rounded, plausible values.

```
# A rough biome guesser from mean annual temperature (C) and rainfall (mm). Illustrative
def biome(temp, rain):
    if temp < -5:
        return "tundra"
    if rain < 250:
        return "desert"
    if temp < 5:
        return "boreal forest"
    if temp < 20:
        return "temperate grassland" if rain < 500 else "temperate forest"
    return "savanna" if rain < 1500 else "tropical rainforest"
places = [
    ("Inverness, Scotland", 9, 700),
    ("Manaus, Brazil", 27, 2300),
    ("Cairo, Egypt", 22, 25),
    ("Serengeti plains", 22, 800),
    ("Kansas prairie", 12, 450),
    ("Arctic coast of Siberia", -12, 200),
    ("Northern Canada", 1, 450),
]
for name, temp, rain in places:
    print(f"{name:24s} {temp:4d} C {rain:5d} mm  ->  {biome(temp, rain)}")

# Output:
# Inverness, Scotland         9 C   700 mm  ->  temperate forest
# Manaus, Brazil             27 C  2300 mm  ->  tropical rainforest
# Cairo, Egypt               22 C    25 mm  ->  desert
# Serengeti plains           22 C   800 mm  ->  savanna
# Kansas prairie             12 C   450 mm  ->  temperate grassland
# Arctic coast of Siberia   -12 C   200 mm  ->  tundra
# Northern Canada             1 C   450 mm  ->  boreal forest
```

Inverness and Manaus come out as expected, and the function separates two warm places by rain: the Serengeti plains and Manaus differ by only 5 degrees but about 1,500 mm of rain, and that is what changes savanna into rainforest. Northern Canada, at 1 °C and 450 mm, lands in the boreal forest. The model also shows its weakness: Cairo and the Arctic coast are both "low rainfall" places but get different labels only because one is below the temperature cut-off. The thresholds are hard lines, whereas in nature one biome fades into the next across a wide transition zone, called an **ecotone**, such as the band where boreal forest thins into tundra.

It Is Not Just How Much Rain, but When

Two places can have the same yearly rainfall and yet be very different biomes, because the *timing* matters. A savanna and a rainforest might both receive over a metre of rain, but the savanna gets it in one season and then dries out for months. Plants and animals there must survive the dry season with deep roots, shed leaves or migration. Ecologists draw a **climate diagram** (often called a Walter-Lieth diagram) with the twelve months along the bottom, and use a handy rule: a month counts as **dry** when its rainfall in millimetres is less than twice its mean temperature in degrees Celsius. The rule reflects that warm air evaporates more water, so the same rain goes less far when it is hot.

```
# Walter-Lieth rule of thumb: a month is dry if its rain (mm) is below twice its mean te
# Monthly figures are rounded, invented-but-plausible values for three climates, January
climates = {
    "Athens-like (Mediterranean)": (
        [10, 11, 13, 17, 22, 26, 29, 29, 25, 20, 15, 11],
        [55, 40, 40, 25, 20, 8, 5, 5, 15, 50, 60, 65]),
    "Fort William-like (west Highland)": (
        [3, 3, 5, 7, 10, 13, 14, 14, 12, 9, 5, 3],
        [250, 200, 180, 120, 100, 100, 120, 140, 180, 230, 240, 260]),
    "Mumbai-like (monsoon)": (
        [24, 25, 27, 29, 30, 29, 27, 27, 27, 28, 27, 25],
        [3, 2, 2, 1, 15, 520, 710, 430, 300, 70, 15, 3]),
}
for name, (temps, rains) in climates.items():
    dry = [rain < 2 * temp for temp, rain in zip(temps, rains)]
    pattern = "".join("D" if d else "." for d in dry)
    print(f"{name}")
    print(f"  mean {sum(temps) / 12:.1f} C, total {sum(rains)} mm, dry months {sum(dry)}
```

```
# Output:
# Athens-like (Mediterranean)
#   mean 19.0 C, total 388 mm, dry months 6  [...DDDDDD...]
# Fort William-like (west Highland)
#   mean 8.2 C, total 2120 mm, dry months 0  [...........]
# Mumbai-like (monsoon)
#   mean 27.1 C, total 2071 mm, dry months 7  [DDDDD.....DD]
```

The Athens-like climate has six dry months, all in the hot summer, which is exactly the Mediterranean pattern: wet mild winters, dry hot summers, and the fire-adapted shrubland shown in the biome table. The Mumbai-like monsoon climate has seven dry months, with the rain arriving as a flood from June to September; wet enough in total to look like a rainforest on paper, but with a long dry season that real rainforest does not have. The west Highland climate has no dry month at all, and with over 2,000 mm of rain a year and mean temperatures of about 8 °C, it supports temperate rainforest-like conditions: mosses, ferns and lichens cover the trees in the Atlantic oakwoods of the west coast. A yearly total can hide the one thing that decides what lives in a place.

A Quick Tour with Nutrient Cycles in Mind

Each biome connects to the earlier chapters. The energy that reaches producers (Ecology and Ecosystems Fundamentals 2) depends on temperature and water, and the way nutrients cycle (Ecology and Ecosystems Fundamentals 4) depends on how fast decomposers can work.

- **Tundra:** cold keeps decomposers slow, so dead plant matter piles up and a great deal of carbon sits frozen in permafrost soil. As the ground warms, that carbon becomes a potential source of carbon dioxide and methane. High Scottish summits, such as the plateau of the Cairngorms, have small patches of this arctic-alpine habitat.
- **Boreal forest:** the largest land biome. Needles decompose slowly into acidic soil, and conifers keep their leaves through the winter so they can start photosynthesis the moment it warms. The Flow Country’s blanket bog (Ecology and Ecosystems Fundamentals 4) is a Scottish cousin with a similar slow-decay, carbon-storing character.
- **Temperate forest:** deciduous trees drop their leaves in autumn, and the leaf litter feeds decomposers and recycles nutrients annually. Much of Britain was once covered by this

biome, but centuries of clearing, grazing and drainage have removed most of it; the remnants, such as the Caledonian pinewood of Strathspey, are carefully protected.

- **Grassland:** fire, grazing and a dry season discourage trees. Grasses put much of their growth below ground, so grassland soils are deep and rich, which is one reason the great prairies became farmland.
- **Savanna:** a mix of trees and grass held in balance by fire, rainfall and herbivores such as elephants that knock trees down. Remove the fires or the grazers and woodland can take over.
- **Tropical rainforest:** warmth and water allow year-round growth, and decomposition is so fast that most of the nutrients are held in the living plants rather than in the soil. That is why the soil under many rainforests is surprisingly poor, and why a cleared patch of forest can only be farmed for a few years before its fertility drops.
- **Desert:** water is the limit. Production is low and patchy, but it comes in bursts after rain; many desert plants spend most of their lives as seeds.

CLIMBING A MOUNTAIN IS LIKE TRAVELLING NORTH

Temperature falls with height, about 6 °C for every 1,000 m on average. Walk up a Scottish hill such as Ben Nevis (1,345 m) and you pass through the same sequence as a journey towards the pole: woodland or farmland at the bottom, then moorland and bog, then tough alpine heath, then bare rock and moss. A **treeline**, above which it is too cold and windy for trees, forms at a few hundred metres in Scotland but much higher in warmer places. Mountains on the wet western side also create a **rain shadow**: air forced up loses its moisture as rain on the western slopes, leaving the east drier, which is why Fort William is far wetter than places only 150 km to the east.

The Ocean and Fresh Water

About 71% of the Earth's surface is ocean, so any tour that stops at the shore leaves out most of the planet. Marine biomes are shaped by **light** (sunlight fades within a couple of hundred metres, so photosynthesis happens only near the surface), **nutrients** (often scarce in the sunlit layer because dead material sinks) and **temperature**. The open ocean is vast but, per square metre, close to a desert: producers are microscopic **phytoplankton**, and the water is often short of nitrogen, phosphorus or iron. Production is far higher where nutrients are brought to the surface, as in coastal **upwelling** zones off Peru and Namibia, or are washed in by rivers, as in estuaries and shallow seas. Coral reefs are an interesting exception: they thrive in clear, nutrient-poor tropical water, recycling nutrients extremely efficiently between coral and algae.

Fresh water is a small share of the planet but a large share of life. Lakes, rivers and wetlands link directly to Ecology and Ecosystems Fundamentals 4: a loch's residence time and phosphorus supply determine whether it is clear and nutrient-poor or green and eutrophic. Wetlands, from Scottish bogs to tropical swamps, are among the most productive and most carbon-rich ecosystems on land.

Which biomes do the most photosynthesis? The last snippet combines a rough area with a rough **net primary production** (NPP, the energy-rich plant matter produced after the plants have paid their own respiration costs, expressed here as grams of carbon per square metre per year). Multiply the two, convert the units, and rank the results. One million square kilometres is 10^{12} m², and a gigatonne of carbon is 10^{15} g, so the conversion is area (in millions of km²) times NPP, divided by 1,000. The list is limited to these eight biomes, so the percentages are shares of those eight only.

```
# Which biomes do the most photosynthesis? Rounded, illustrative figures only.
# (area in million km2, net primary production in g of carbon per m2 per year)
biomes = {
    "tropical rainforest": (17, 900),
    "savanna": (15, 400),
    "temperate forest": (12, 600),
    "boreal forest": (12, 250),
    "temperate grassland": (9, 300),
    "tundra": (8, 100),
    "desert": (45, 40),
    "open ocean": (330, 60),
}
total_area = sum(area for area, npp in biomes.values())
totals = {name: area * npp / 1000 for name, (area, npp) in biomes.items()}   # GtC a yea
grand = sum(totals.values())
print(f"{'biome':20s} {'area %':>7s} {'GtC/yr':>7s} {'NPP %':>6s} {'g C/m2':>7s}")
for name in sorted(totals, key=totals.get, reverse=True):
    area, npp = biomes[name]
    print(f"{name:20s} {area / total_area:7.1%} {totals[name]:7.1f} {totals[name] / gran
print(f"all eight biomes: {grand:.1f} GtC a year")

# Output:
# biome                area %  GtC/yr  NPP %  g C/m2
# open ocean            73.7%    19.8  35.0%      60
# tropical rainforest    3.8%    15.3  27.0%     900
# temperate forest       2.7%     7.2  12.7%     600
# savanna                3.3%     6.0  10.6%     400
```

```
  # boreal forest           2.7%     3.0   5.3%     250
  # temperate grassland     2.0%     2.7   4.8%     300
  # desert                 10.0%     1.8   3.2%      40
  # tundra                  1.8%     0.8   1.4%     100
  # all eight biomes: 56.6 GtC a year
```

The ocean covers nearly three quarters of the area in this list but contributes only 35% of the production, because each square metre makes so little (60 g C compared with 900 for rainforest). Tropical rainforest is the opposite: under 4% of the area, yet over a quarter of the production. Desert and tundra are cheap to hold on to in area terms and nearly irrelevant in production terms. This is why losing a small area of rainforest matters more to the planet's plant production than losing the same area of desert would, an idea Exercise 3 puts a number on. The exact figures vary widely between sources, but the pattern, rainforest and ocean at the top and cold or dry biomes at the bottom, is robust.

Biomes Are Not Fixed

Biome maps look permanent, but they describe the climate of a particular period. Ten thousand years ago much of Scotland was under or just beyond ice, and the vegetation advanced northwards as the climate warmed, from tundra to birch and pine to broadleaf forest. Today a warming climate is moving the zones again: treelines creep uphill, boreal forest edges push north into tundra, and dry biomes expand in some regions. Biomes shift only as fast as seeds can spread and trees can grow, so the vegetation tends to lag behind the climate, and cleared, farmed or drained land is held in a human-made state. Exercise 1 explores the direction of these shifts with the biome function.

	TUNDRA	TEMPERATE FOREST	TROPICAL RAINFOREST
Growing season	Weeks	Several months	All year
Decomposition	Very slow; carbon stored	Moderate; yearly leaf-fall cycle	Very fast; nutrients held in living plants
Where most carbon sits	In frozen soil	In trees and soil	In trees
Production per m²	Low	High	Highest on land

Common mistakes

Do not define a desert by heat: it is defined by dryness, and some are bitterly cold. Do not judge a climate by the yearly rainfall total alone, since the timing of the rain decides how long the dry season is. Do not assume biome boundaries are sharp lines on a map: they are fuzzy transition zones, and the cut-offs in this chapter's function are a teaching device. Do not say the rainforest has rich soil just because it has lush growth; the nutrients are in the trees. And do not confuse a biome (a type of ecosystem, defined by climate and vegetation) with a habitat (the home of one species) or with a single ecosystem, such as one particular loch.

Challenges

EXERCISE 1

Reuse the `biome(temp, rain)` function. Take two sites, a highland glen at 3 °C with 700 mm of rain and a mild lowland site at 17 °C with 600 mm, and print the biome for each at warming of 0, 2, 4 and 6 °C with rainfall held fixed. Say which biome changes first at each site, and name two reasons real change would be slower and messier than your printout.

View solution

EXERCISE 2

Two invented climates, January to December. Cape Town-like: temperatures `[21, 21, 20, 17, 15, 13, 12, 13, 14, 16, 18, 20]`, rainfall `[15, 17, 20, 41, 69, 93, 82, 77, 40, 30, 20, 17]`. Edinburgh-like: temperatures `[4, 4, 6, 8, 11, 14, 16, 15, 13, 9, 6, 4]`, rainfall `[50, 40, 40, 40, 50, 55, 70, 65, 60, 65, 60, 55]`. Using the dry-month rule, list the dry months of each as month numbers, and label each as having no dry season (none), a short one (1 to 3 months) or a long one (4 or more). Which one fits Mediterranean shrubland?

View solution

EXERCISE 3

Use the production table without the ocean (seven land biomes) and work out the land total in GtC a year. Then clear 2 million km² of tropical rainforest for pasture that produces 350 g C per m² per year, and print the total before and after, the change in GtC and as a percentage, and the loss per square

metre cleared. Add a sentence on what this sum leaves out about the carbon already stored in the forest.

View solution

QUICK REFERENCE

- **Biome:** a large region defined by its climate and typical vegetation, whatever the exact species
- **Main drivers:** mean annual temperature and precipitation; also soil, fire, grazing and altitude
- **Desert:** defined by low precipitation (under about 250 mm a year), not by heat
- **Dry month (Walter-Lieth rule):** rainfall in mm is less than twice the mean temperature in °C
- **Timing of rain** can matter as much as the yearly total: savanna and monsoon climates have long dry seasons
- **Production:** highest in tropical rainforest per square metre; the ocean makes a large share of the total only because it is so big
- **Altitude:** climbing about 1,000 m is like travelling several degrees of latitude towards the pole
- **Ecotone:** the blurry transition zone between two biomes

COMING NEXT

Ecology and Ecosystems Fundamentals 6 zooms in from the planet to a single species: how populations grow, what limits them, and the curves that describe them.

Ecology and Ecosystems Fundamentals

Course 1 · Chapter 6 · Populations

Chapter 5 toured the biomes and ended with a puzzle: if climate decides how much plant matter a place can produce, what decides how many animals, or how many trees, or how many bacteria, the place can actually hold? To answer that we zoom in from the planet to a single species. A **population** is all the individuals of one species living in one area at one time: the red deer of a Highland estate, the frogs in one pond, the oak trees in a wood. Ecologists care about populations because they are what conservation, farming and pest control actually deal with. This chapter looks at how populations change, the two classic growth curves, and what finally holds a population back. The maths stays light: addition, multiplication and one short formula, always run in Python.

WHAT TO TRUST IN THIS CHAPTER

The Python snippets here and in the three exercise solutions were run, and the output in the comments is what Python printed. The rates, starting numbers and carrying capacities are *invented round figures chosen for practice*, not measurements of real species. The ecological background is written from general knowledge, not fetched from a source for this chapter, so check any figure you want to rely on, such as the island reindeer numbers mentioned below, against a current textbook or a national agency.

Four Ways a Population Changes

A population goes up or down for only four reasons. Individuals are added by **births** and **immigration** (arrivals from elsewhere) and removed by **deaths** and **emigration** (departures). Written as a sum, over one time step:

New N = N + births + immigration − deaths − emigration

On an island or in a closed pond, immigration and emigration are small, so births and deaths do most of the work. Ecologists usually express them *per head*, as a **per capita rate**: if a population of 100 produces 30 young a year, the per capita birth rate is 0.30. If 10 die, the per capita death rate is 0.10. The difference, **r = births per head − deaths per head**, is the **net growth rate**. A positive r means the population grows, a negative r means it shrinks and r = 0 means it is steady. The whole subject is about what makes r change.

Exponential Growth: Unlimited Room

Imagine a few deer released on a large, empty island with plenty of food and no predators. Each year the population grows by r times its current size. Because the growth is a *fraction of what is already there*, bigger populations add bigger numbers, which makes them grow even faster. That runaway pattern is **exponential growth**, and the same maths governs a bacterial colony in a dish, a viral post being shared and money earning compound interest.

```
# Exponential growth: each year N grows by births minus deaths per head. Invented but pl
import math
births, deaths = 0.30, 0.10          # per animal per year
r = births - deaths                  # net growth rate per head per year
n = 50                               # starting population (say, deer introduced to a ne
print("year  population")
for year in range(0, 21, 4):
    print(f"{year:4d}  {n:10.0f}")
    for _ in range(4):
        n = n + r * n
doubling = math.log(2) / math.log(1 + r)
print(f"net rate r = {r:.2f} per head per year; doubling time = {doubling:.1f} years")

# Output:
# year  population
#    0          50
#    4         104
#    8         215
#   12         446
#   16         924
#   20        1917
# net rate r = 0.20 per head per year; doubling time = 3.8 years
```

Fifty animals become 104 after four years and more than 1,900 after twenty, and the numbers added grow steadily larger. The **doubling time** of 3.8 years stays the same wherever you start: 50 to 100 takes as long as 1,000 to 2,000. That is why exponential growth is deceptive. For the first decade nothing looks alarming, then in the next few years the population explodes. Exponential growth is real but never lasts, because the island, its food and its space are finite.

A BOOM AND A CRASH

Real populations sometimes show the boom stage and what follows. A famous case is a herd of reindeer released on a remote Bering Sea island in the 1940s, where, by the standard account, a few dozen animals multiplied into thousands within about twenty years and then crashed, leaving only a handful. The pattern, a fast rise followed by overshoot and collapse, is a warning of what happens when a population outruns its food supply. Closer to home, the Soay sheep of the St Kilda islands off north-west Scotland have long been studied for their boom and bust years, when a mild winter lets numbers climb and a hard one with little food cuts them back sharply. Check the details of both against a current source before quoting them.

Logistic Growth: Carrying Capacity

Nothing grows exponentially for ever. As a population crowds the available food, space and shelter, each animal has fewer resources, so births per head fall, deaths per head rise, or both. The largest population an area can support indefinitely is its **carrying capacity**, written **K**. It is not a fixed number of the species: it depends on the habitat, the season and even the year (see Ecology and Ecosystems Fundamentals 5 for how climate sets the underlying production).

The simplest model that includes K multiplies the exponential growth by a brake, `(1 - N / K)`. When N is small the brake is close to 1 and growth is almost exponential. When N reaches K the brake is 0 and growth stops. This is **logistic growth**, and it draws the famous S-shaped curve.

```
# Logistic growth: growth slows as the population N approaches the carrying capacity K.
r, K = 0.5, 1000
n = 20.0
print("year  population  growth that year")
for year in range(0, 21):
    growth = r * n * (1 - n / K)
    if year % 2 == 0:
        print(f"{year:4d}  {n:10.0f}  {growth:10.1f}")
    n = n + growth

# Output:
# year  population  growth that year
#    0          20         9.8
#    2          44        21.1
#    4          96        43.4
#    6         199        79.8
```

```
#    8          380       117.8
#   10          622       117.5
#   12          836        68.5
#   14          948        24.7
#   16          986         7.0
#   18          996         1.8
#   20          999         0.5
```

Read down the right-hand column. Growth is small when the population is tiny (9.8 animals in the first year), peaks at 117.8 a year around year 8 when N is 380, and then fades away to almost nothing as N creeps up to 999. The speed is greatest at about half the carrying capacity, here K/2 = 500, and the peak growth is roughly r×K/4 = 125. A population does not hit K with a bump; it eases into it, because the brake gets gentler as the room runs out.

	EXPONENTIAL GROWTH	LOGISTIC GROWTH
Shape of curve	J-shaped, climbing ever faster	S-shaped, levelling off at K
Resources	Unlimited	Limited; shared by a growing population
Growth per head	Stays at r	Falls towards 0 as N approaches K
Where it fits	Early colonisation, a new empty habitat	Established populations in a fixed area
Real example	Bacteria in fresh broth; an invading species at first	Fish in a stocked pond; trees in a growing woodland

What Limits a Population?

The brake in the logistic model stands for many real causes. Ecologists divide them into two kinds.

- **Density-dependent factors** get stronger as the population gets denser: competition for food and space, disease spreading between crowded animals, and predators that find a plentiful prey easier to catch. These are the true regulators, because they push a crowded population down and a sparse one up.
- **Density-independent factors** hit a population by the same fraction, however crowded it is: a severe winter, a flood, a wildfire, a drought. They can knock numbers down without any link

to how full the habitat was. Soay sheep crashes are often triggered by weather, though crowding makes the effect worse.

Density dependence is the idea that holds the logistic model together, and Exercise 3 shows it directly: when birth and death rates change with density, the population settles at a steady value by itself. It also ties back to earlier chapters. The energy a population can use is limited by how much reaches its level of the food chain (Ecology and Ecosystems Fundamentals 2), and the predators and prey around it form the web described in Ecology and Ecosystems Fundamentals 3.

Life Strategies: Many Cheap Young or a Few Costly Ones

Species do not all balance births and deaths in the same way. Two ends of a scale are often described. **r-selected** species live in unstable or newly opened habitats and reproduce quickly: many small offspring, little care, short lives. **K-selected** species live near the carrying capacity of stable habitats and invest heavily in few young. Most species lie somewhere between, so treat these as two ends of a spectrum rather than two boxes.

	R-SELECTED (FAST, MANY YOUNG)	K-SELECTED (SLOW, FEW YOUNG)
Offspring	Very many, small	Few, large, well cared for
Lifespan and maturity	Short; mature early	Long; mature late
Typical habitat	Disturbed, changeable, new	Stable, crowded
Typical examples	Bacteria, weeds, aphids, mice	Elephants, whales, oaks, humans

This matters for people. A pest with a high r, such as an aphid or a rat, can bounce back quickly after control. A slow-breeding animal such as a whale, an albatross or a tortoise takes decades to recover from over-hunting, which is why rare large species need so much careful protection. Age matters too: a population full of young animals is about to grow and one made mostly of old animals is about to shrink, so ecologists draw **age structure** diagrams, often as a pyramid, to judge the future of a population.

Harvesting a Population

If a population is growing fastest around K/2, then taking animals out at that point does the least damage, because the population replaces them quickest. That is the idea behind **maximum sustainable yield** in fishing, forestry and deer management: the most that can be removed every year without shrinking the stock. In the logistic model it is about r×K/4. The snippet tries different yearly harvests and shows what is left after sixty years.

```
# Harvesting a logistic population: how much can be taken each year without it collapsing
r, K = 0.5, 1000
def final_population(harvest, years=60):
    n = 500.0
    for _ in range(years):
        n = n + r * n * (1 - n / K) - harvest
        if n <= 0:
            return 0.0
    return n
print("harvest per year  population after 60 years")
for harvest in (0, 50, 100, 125, 130, 150):
    print(f"{harvest:16d}  {final_population(harvest):10.0f}")
print(f"theoretical maximum sustainable yield = r*K/4 = {r * K / 4:.0f}")

# Output:
# harvest per year  population after 60 years
#                0        1000
#               50         887
#              100         724
#              125         500
#              130           0
#              150           0
# theoretical maximum sustainable yield = r*K/4 = 125
```

With no harvest the population settles at K = 1,000. Taking 50 a year leaves 887, taking 100 leaves 724, and taking 125, exactly the theoretical maximum, holds the population at 500, the point of fastest growth. But a harvest of 130, only five a year more, drives the population to zero within sixty years. The cliff edge is the lesson. Fish stocks have collapsed in exactly this way, partly because the true carrying capacity and growth rate are never known precisely, so a quota set at the “maximum” leaves no margin for a bad year. Managers therefore aim well below the maximum yield.

Common mistakes

Do not think a population grows exponentially for ever: the exponential curve describes only the early stage, before limits bite. Do not treat carrying capacity as a fixed number for a species: it changes with habitat, season and year, and a population can overshoot it and crash. Do not mix up density-dependent factors (strengthen with crowding) and density-independent ones (hit regardless, like a storm). Do not assume that the biggest harvest is the best: harvesting at the theoretical maximum is dangerously close to collapse. And do not confuse a population (one species in one place) with a community (all the species of a place), which is the subject of the next chapters.

Challenges

EXERCISE 1

An invasive species starts with 100 individuals and has a per capita birth rate of 0.55 and death rate of 0.15 per year, with no limits. Use a loop to find the first year the population passes 10,000, print the net rate r and the doubling time (`math.log(2) / math.log(1 + r)`), and print what the population would reach three years after that. Say in a sentence why this is the stage when control is easiest to miss.

View solution

EXERCISE 2

Run logistic growth for 40 years with K = 500, a starting population of 10, and two growth rates, r = 0.8 and r = 0.2. For each, find the year in which growth per year is greatest, the population then and the population after 40 years. What does the peak position tell you about where growth is fastest, and what does a higher r change?

View solution

EXERCISE 3

Make the rates density-dependent: births per head `b = 0.40 - 0.0002 * N` and deaths per head `d = 0.10 + 0.0001 * N` , with each year's change equal to `(b - d) * N` . First work out by algebra, and print, the population at which b = d. Then simulate 30 years from a start of 100 and from a start of 1,500, printing the population every 10 years. What does this show about carrying capacity?

View solution

QUICK REFERENCE

- **Population:** all the individuals of one species in one area at one time
- **Change:** births + immigration − deaths − emigration; per capita r = birth rate − death rate
- **Exponential growth:** N grows by r×N each step, giving a J-shaped curve with a constant doubling time of ln 2 / ln(1 + r)
- **Logistic growth:** growth = r×N×(1 − N/K); S-shaped curve levelling off at the carrying capacity K
- **Fastest growth** at about K/2, where the rate is roughly r×K/4
- **Density-dependent:** competition, disease, predation (strengthen with crowding); **density-independent:** storms, floods, fire
- **r-selected:** many cheap young, fast; **K-selected:** few costly young, slow
- **Maximum sustainable yield** is about r×K/4, but harvesting right at it leaves no safety margin

COMING NEXT

Ecology and Ecosystems Fundamentals 7 asks what happens when two species want the same resources: competition, niches, and why similar species can sometimes share a habitat and sometimes cannot.

Ecology and Ecosystems Fundamentals

Course 1 · Chapter 7 · Competition and Niches

Chapter 6 treated a population as if it lived alone, held back only by its own crowding. Real habitats are not like that. Several species share every wood, pond and shoreline, and many of them need the same food, space or nesting sites. When a resource is in short supply, the species using it affect each other, and that interaction, **competition**, helps decide who lives where. This chapter introduces the **niche**, the idea that each species has its own way of making a living, shows how competition between two species is modelled, and explains why similar species either divide a habitat between them or one of them disappears. The maths stays light: addition, multiplication and comparisons, always run in Python.

> **WHAT TO TRUST IN THIS CHAPTER**
>
> The Python snippets here and in the three exercise solutions were run, and the output in the comments is what Python printed. The species, diet shares, growth rates and competition strengths are *invented round figures chosen for practice*, not measurements. The ecological background is written from general knowledge, not fetched from a source for this chapter, so check any real-world example you want to rely on, such as the squirrel and tit examples below, against a current textbook or a national agency.

What Is a Niche?

A species' **habitat** is where it lives: a beech wood, a rock pool, a peat bog. Its **niche** is its whole way of life there: what it eats, when it is active, where it feeds and nests, what temperatures and conditions it tolerates, and what eats it. A useful comparison is that the habitat is the address and the niche is the job. Many species can share one address, but each has a different job.

Ecologists separate two ideas. The **fundamental niche** is the full range of conditions and resources a species could use if nothing else were around. The **realised niche** is the smaller range it actually uses once competitors, predators and disease are taken into account. Barnacles on a rocky shore are the classic example: one species can survive across a wide band of the shore (its fundamental niche) but a rival pushes it out of the lower part, so it is only found higher up (its realised niche). The niche is therefore not a place on a map. It is a position along many axes at once, such as food size, feeding height and season.

	HABITAT	NICHE
Meaning	The place where a species lives	The species' whole way of life and role
Question it answers	Where is it found?	What does it do, and how, and when?
Shared?	Yes: many species share one habitat	Partly: species can overlap, but never completely for long
Example	An oak wood in southern England	A tit that feeds on caterpillars in the high branches in spring

Competition: Two Kinds

Competition happens when two or more individuals use the same limited resource and each lowers the other's survival or reproduction. Chapter 6 covered competition *within* a species, the density-dependent brake that creates carrying capacity (see Ecology and Ecosystems Fundamentals 6). This chapter is about competition *between* species. It comes in two main forms.

- **Exploitative competition:** species use up a shared resource without ever meeting. Two seed-eating birds that feed in the same field simply leave fewer seeds for each other, and so do two plants whose roots draw on the same water.
- **Interference competition:** one species directly blocks the other, by fighting, defending a territory, shading or poisoning. Plants that shade out their neighbours, and birds that drive rivals from a feeder, are examples.

Competition costs both sides, which is why ecologists describe it as a (−, −) interaction. The costs are rarely equal, though: usually one species is hurt more than the other, and that imbalance decides the outcome.

Measuring Niche Overlap

If niche overlap drives competition, we can estimate how strong it is. One simple measure takes the share of each species' resource use that falls into each category, and adds up the smaller of the two shares category by category. The total runs from 0, meaning the species never use the

same resource, to 1, meaning they use it in identical proportions. The snippet compares three invented finches by the seed size they eat.

```
# Niche overlap: share of each species' food (by seed size) that the two species both us
sizes = ["tiny", "small", "medium", "large", "huge"]
use = {
    "finch A": [0.05, 0.30, 0.45, 0.15, 0.05],
    "finch B": [0.00, 0.10, 0.35, 0.40, 0.15],
    "finch C": [0.60, 0.35, 0.05, 0.00, 0.00],
}
def overlap(a, b):
    return sum(min(x, y) for x, y in zip(use[a], use[b]))
names = list(use)
print("pair              overlap")
for i in range(len(names)):
    for j in range(i + 1, len(names)):
        print(f"{names[i]} + {names[j]}   {overlap(names[i], names[j]):.2f}")

# Output:
# pair              overlap
# finch A + finch B   0.65
# finch A + finch C   0.40
# finch B + finch C   0.15
```

Finches A and B share most of the middle-sized seeds, so their overlap is high (0.65) and they are the pair most likely to compete. Finches B and C eat at opposite ends of the size range, so the overlap is only 0.15 and they hardly affect one another. High overlap does not by itself prove competition, because it only matters if the shared resource is actually scarce. When seeds are abundant, two finches can share them without harm.

Competitive Exclusion

What happens when two species overlap almost completely? In the 1930s the Russian biologist Georgyi Gause grew two species of *Paramecium*, a single-celled pond organism, in test tubes with a fixed daily supply of food. Each species thrived alone, but when the two were mixed, one species eventually died out. This result is known as the **competitive exclusion principle**: *two*

species that depend on exactly the same limiting resource cannot coexist for ever; the slightly better competitor takes over. It is sometimes put as "complete competitors cannot coexist".

A wild version may be familiar from Britain. In many English and Welsh woods and in parts of Scotland, the introduced grey squirrel has replaced the native red squirrel, which has retreated to places such as Highland pinewoods and some islands. The usual account is that greys use a similar diet and habitat, cope better with acorns and can carry a virus that harms reds. That is a mix of competition and disease, so it is not a pure textbook case, and the details are worth checking against a current source.

Modelling Two Competitors

We can extend the logistic model from Chapter 6 so that each species is slowed by *both* populations. Species 1 grows as `r1 × N1 × (1 − (N1 + a12 × N2) / K1)`, and species 2 likewise. The new quantity `a12` is the **competition coefficient**: how much one individual of species 2 slows species 1, measured in units of a species 1 individual. If a12 = 0.5, each member of species 2 takes up half as much of species 1's resources as one of its own kind would. If a12 = 1.5, it takes more. These are the **Lotka–Volterra competition equations**, named after the two scientists who proposed them in the 1920s. Two simulations follow: weak competition first, then a case where species 2 hurts species 1 more than the reverse.

```
# Two competing species (Lotka-Volterra competition): each one's growth is slowed by BOTH
def simulate(r1, r2, k1, k2, a12, a21, n1, n2, years):
    for year in range(years + 1):
        if year % 20 == 0:
            print(f"{year:4d}  {n1:8.0f}  {n2:8.0f}")
        g1 = r1 * n1 * (1 - (n1 + a12 * n2) / k1)
        g2 = r2 * n2 * (1 - (n2 + a21 * n1) / k2)
        n1, n2 = max(n1 + g1, 0.0), max(n2 + g2, 0.0)
print("A: weak competition (alpha 0.5 both ways)")
print("year  species1  species2")
simulate(0.4, 0.4, 1000, 1000, 0.5, 0.5, 50, 50, 100)
print()
print("B: strong competition (species 2 hurts species 1 more than the reverse)")
print("year  species1  species2")
simulate(0.4, 0.4, 1000, 1000, 1.5, 0.6, 50, 50, 100)

# Output:
```

```
# A: weak competition (alpha 0.5 both ways)
# year  species1  species2
#    0        50        50
#   20       665       665
#   40       667       667
#   60       667       667
#   80       667       667
#  100       667       667
#
# B: strong competition (species 2 hurts species 1 more than the reverse)
# year  species1  species2
#    0        50        50
#   20        75       913
#   40         1       999
#   60         0      1000
#   80         0      1000
#  100         0      1000
```

In run A both species are held below their solo carrying capacity of 1,000 and settle together at 667 each: **stable coexistence**. Each species limits itself (a species' own members count in full) more than it limits the other (rival members count half), so neither can take over. In run B species 2 counts 1.5 times over against species 1, but species 1 counts only 0.6 against species 2. Both start at 50, yet species 1 is driven down to 1 by year 40 and to zero by year 60, while species 2 reaches its full 1,000. That is competitive exclusion in numbers.

Who Wins? A Simple Rule

You do not have to run a simulation to predict the result. Species 1 can increase when rare and species 2 is at its carrying capacity only if `K1 > a12 × K2`, and the same logic applies to species 2. This gives four outcomes, which the next function reports for a few cases.

```
# Which outcome? Compare each species' alpha with the ratio of carrying capacities.
def outcome(k1, k2, a12, a21):
    s1_wins_alone = k1 > a12 * k2     # species 1 can grow when species 2 is at its K
    s2_wins_alone = k2 > a21 * k1
    if s1_wins_alone and s2_wins_alone:
        return "stable coexistence"
```

```
        if s1_wins_alone:
            return "species 1 excludes species 2"
        if s2_wins_alone:
            return "species 2 excludes species 1"
        return "winner depends on who starts ahead"
    cases = [(1000, 1000, 0.5, 0.5), (1000, 1000, 1.5, 0.6), (1000, 1000, 1.2, 1.2), (1000,
    for k1, k2, a12, a21 in cases:
        print(f"K1={k1} K2={k2} a12={a12} a21={a21}: {outcome(k1, k2, a12, a21)}")

    # Output:
    # K1=1000 K2=1000 a12=0.5 a21=0.5: stable coexistence
    # K1=1000 K2=1000 a12=1.5 a21=0.6: species 2 excludes species 1
    # K1=1000 K2=1000 a12=1.2 a21=1.2: winner depends on who starts ahead
    # K1=1000 K2=600 a12=0.5 a21=1.5: species 1 excludes species 2
```

The pattern is the heart of the chapter. Coexistence needs each species to limit *itself* more than it limits its rival. That happens when the species use their resources differently enough, which means their niches must differ. The third case, where each species hurts the other more than itself, is unstable: a small early lead is enough to win, which is why, in a real habitat, the first colonist can sometimes keep a latecomer out.

How Similar Species Share a Habitat

Gause's principle might suggest every habitat should hold only one species per resource, yet a wood holds many insect-eating birds. The answer is **resource partitioning**: similar species divide the habitat so that each uses it differently. Coal, blue and great tits are often cited from British woodland. They are said to forage in different parts of the trees and to take different sizes of food, so competition is weakened. Warblers in a spruce forest are another classic story, feeding at different heights and in different parts of the branches. Check specifics against a current source before quoting them.

Over many generations, competition can even push species apart. **Character displacement** is the evolutionary shift of a trait, such as beak size, in two similar species where they live together, making them more different than where each lives alone. Darwin's finches on the Galapagos Islands are the standard example, and are described that way in most textbooks. The overall lesson is that competition shapes not only numbers but also the differences between species.

	COMPETITIVE EXCLUSION	RESOURCE PARTITIONING
Niche overlap	Almost complete	Partial; species use resources differently
Outcome	One species disappears locally	Both species persist
Competition coefficients	At least one is large (above 1 relative to K)	Both small: each limits itself more than the other
Example	Gause's two *Paramecium* species in a test tube	Several tit or warbler species in one wood

Common mistakes

Do not confuse a habitat (where) with a niche (what and how). Do not assume that overlap alone means competition: it only matters if the shared resource is limited. Do not read competitive exclusion as meaning one species always wins everywhere, since it applies to a specific place and set of conditions, and a species that loses in one habitat can win in another. Do not forget that competition acts on both species, even when one is hurt far more. And remember that real cases, like the squirrels, mix competition with disease, predation and human action, so a single cause is rarely the whole story.

Challenges

EXERCISE 1

Three woodland birds feed in five zones (ground, low, middle, high, canopy) with these shares. Bird X: 0.40, 0.30, 0.20, 0.10, 0.00. Bird Y: 0.05, 0.25, 0.40, 0.25, 0.05. Bird Z: 0.00, 0.05, 0.15, 0.30, 0.50. Write an `overlap` function and print the overlap for every pair, naming the pair most likely to compete. Then change Bird Y to 0.00, 0.10, 0.30, 0.35, 0.25 and repeat. What happened, and what does it say about moving away from one competitor?

View solution

EXERCISE 2

Simulate two competing species with K = 1,000 and r = 0.4 for both, and the same competition coefficient in both directions, for 200 years. Try alpha = 0.8 and alpha = 1.2, each starting from (100, 50) and then from (50, 100). Print the final populations for all four runs. Explain in a sentence or two why the starting numbers matter for one alpha and not the other.

View solution

EXERCISE 3

With K1 = K2 = 1,000, build a table of outcomes for every pairing of a12 and a21 from the values 0.4, 0.8, 1.2 and 1.6, using the rule `K1 > a12 × K2` and its mirror image. Label each square coexist, 1 wins, 2 wins or either, and count the coexistence squares. What do those squares have in common?

View solution

QUICK REFERENCE

- **Habitat:** where a species lives; **niche:** its whole way of life (food, place, timing, conditions)
- **Fundamental niche:** what a species could use alone; **realised niche:** what it uses given competitors
- **Competition:** exploitative (using up a shared resource) or interference (direct blocking); a (−, −) interaction
- **Niche overlap:** sum of the smaller shares, from 0 (none) to 1 (identical)
- **Competitive exclusion:** complete competitors cannot coexist for ever; the better one takes over
- **Competition coefficient (a12):** how strongly one individual of species 2 limits species 1, compared with one of its own kind
- **Coexistence rule:** K1 > a12×K2 and K2 > a21×K1; each species limits itself more than its rival
- **Resource partitioning** and **character displacement** let similar species share a habitat

COMING NEXT

Ecology and Ecosystems Fundamentals 8 moves from species that compete to species that eat, help or live on one another: predator and prey cycles, mutualism and parasitism.

Ecology and Ecosystems Fundamentals

Course 1 · Chapter 8 · Predation and Cooperation

Chapter 7 looked at species that harm each other by sharing a limited resource. Species affect one another in many other ways. Some eat each other, some live on each other, and some help each other so much that neither does well alone. This chapter surveys the main kinds of interaction, then looks closely at three: **predation**, where the numbers of eater and eaten rise and fall in linked cycles; **mutualism**, where both partners benefit; and **parasitism**, where one species gains at the other's expense without usually killing it quickly. The maths stays light: a few growth and loss terms added together, always run in Python.

WHAT TO TRUST IN THIS CHAPTER

The Python snippets here and in the three exercise solutions were run, and the output in the comments is what Python printed. The rates, starting numbers and benefits are *invented round figures chosen for practice*, not measurements. The ecological background is written from general knowledge, not fetched from a source for this chapter, so check any real-world example you want to rely on, such as the hare and lynx records or the rabbit disease example below, against a current textbook or a national agency.

Six Ways to Interact

Ecologists describe an interaction by its effect on each partner: a plus sign (+) if the species gains, a minus sign (−) if it loses, and a zero (0) if it is not noticeably affected. Chapter 7's competition is (−, −). The full set of pairings is below.

INTERACTION	EFFECTS	EXAMPLE
Competition	− / −	Two finch species using the same seeds
Predation	+ / −	A fox killing and eating a rabbit
Parasitism	+ / −	A tick feeding on a deer
Mutualism	+ / +	A bee feeding on nectar while pollinating the flower
Commensalism	+ / 0	Barnacles riding on a whale, which is barely affected

INTERACTION	EFFECTS	EXAMPLE
Amensalism	0 / −	Cattle trampling a patch of grass without gaining anything from it

The signs are a useful first sort, but not a final verdict. The same pair can change category with conditions: a fungus that helps a tree in poor soil may become a drain on it when the tree has plenty of nutrients. The snippet below turns the sign table into a tiny classifier. Notice that predation and parasitism share the same signs, so the code cannot tell them apart. The difference is one of lifestyle, which the next sections explain.

```
# Name an interaction from its effects on each species: + helps, - harms, 0 no effect.
def kind(effect1, effect2):
    pair = (effect1, effect2)
    if pair == ("+", "+"): return "mutualism"
    if pair == ("-", "-"): return "competition"
    if "0" in pair and "+" in pair: return "commensalism"
    if "0" in pair and "-" in pair: return "amensalism"
    if "+" in pair and "-" in pair: return "predation or parasitism"
    return "neutral"
examples = [
    ("bee and flower", "+", "+"),
    ("two finches, same seeds", "-", "-"),
    ("barnacle on a whale", "+", "0"),
    ("fox and rabbit", "+", "-"),
    ("tick and deer", "+", "-"),
    ("cattle trampling grass", "0", "-"),
]
for name, e1, e2 in examples:
    print(f"{name:26} ({e1},{e2}) -> {kind(e1, e2)}")

# Output:
# bee and flower             (+,+) -> mutualism
# two finches, same seeds    (-,-) -> competition
# barnacle on a whale        (+,0) -> commensalism
# fox and rabbit             (+,-) -> predation or parasitism
# tick and deer              (+,-) -> predation or parasitism
# cattle trampling grass     (0,-) -> amensalism
```

Predation and Predator–Prey Cycles

A **predator** kills and eats other animals (its **prey**). Herbivores eating plants are often treated in the same way, which is why this chapter uses "predation" loosely for any case where one organism feeds on another. From Chapter 3 you already know that predators can shape whole food webs. Here the question is narrower: how do the *numbers* of predator and prey change together?

Imagine rabbits and foxes on an island. With few foxes the rabbits multiply. More rabbits means more food, so the foxes multiply too. More foxes then eat too many rabbits, the rabbits crash, the foxes go hungry and decline, and with few foxes the rabbits can recover. The result is a repeating cycle in which the predator curve follows the prey curve a little later.

The **Lotka–Volterra predator–prey equations**, proposed independently by Alfred Lotka and Vito Volterra in the 1920s, capture this with two lines. Prey gain `a × prey` from their own births and lose `b × prey × predators` to capture, which rises when either is more common. Predators gain `c × prey × predators` from eating and lose `d × predators` through ordinary deaths. We step through time in tiny jumps of 0.01 years so the simple method stays reasonably accurate.

```
# Predator and prey cycles (Lotka-Volterra), small time steps. Invented round figures.
prey, pred = 40.0, 20.0
a, b, c, d = 0.6, 0.02, 0.01, 0.5   # prey growth, capture rate, conversion, predator de
dt = 0.01
print("year   prey  predators")
for step in range(0, 6001):
    if step % 300 == 0:
        print(f"{step * dt:4.0f}  {prey:5.1f}  {pred:5.1f}")
    new_prey = prey + (a * prey - b * prey * pred) * dt
    new_pred = pred + (c * prey * pred - d * pred) * dt
    prey, pred = new_prey, new_pred

# Output:
# year   prey  predators
#    0   40.0   20.0
#    3   72.7   23.7
#    6   54.6   44.6
#    9   29.9   31.3
#   12   42.7   19.3
#   15   75.8   25.5
#   18   49.5   45.2
```

```
# 21  29.6  29.3
# 24  45.7  18.8
# 27  78.0  27.8
# 30  44.8  45.1
# 33  29.6  27.4
# 36  49.0  18.4
# 39  79.1  30.3
# 42  40.6  44.3
# 45  29.9  25.6
# 48  52.6  18.3
# 51  78.8  33.2
# 54  37.1  43.1
# 57  30.6  24.1
# 60  56.4  18.4
```

Read down the columns. The prey rise first, and the predators peak a couple of years later, as their numbers catch up with the extra food. Each full cycle takes roughly twelve years. You will also notice that each swing is slightly bigger than the last (the prey reading at years 15, 27 and 39 is about 76, 78 and 79). That widening is a side effect of the simple stepping method: the exact equations would repeat the same loop for ever, which is also unrealistic, because the model has no limit on prey growth. Real populations are held in check by food and space, and a better model adds a carrying capacity for the prey, as in Chapter 6.

The classic wild example is the snowshoe hare and the Canada lynx, whose numbers recorded in the fur trade rose and fell together in cycles of roughly ten years. Later work suggested that the hares' own food supply matters as well as the lynx, so it is not a pure two-species story. Rabbits in Britain give another lesson: after the myxomatosis virus was introduced in the 1950s, rabbit numbers crashed, and so did the numbers of some animals that depended on them. Check the details against a current source before quoting them.

Mutualism: Both Sides Gain

In a **mutualism**, each species does better with its partner than without it. The partnership is a trade: one side provides food, shelter or transport, and the other provides something in return.

- **Pollination:** bees, hoverflies and moths collect nectar and carry pollen from flower to flower. Plants gain seed, and the insects gain food.

- **Mycorrhizae:** fungi attached to tree roots extend the root system and bring in water and minerals, while the tree supplies sugars from photosynthesis. Most land plants, including the pines and birches of Scottish woods, have such partners.
- **Lichens:** a fungus and an alga (or a bacterium) living as one. The alga makes food from light, and the fungus provides structure and protection.
- **Cleaner fish:** small fish remove parasites and dead skin from larger fish on coral reefs, getting a meal and giving a cleaning service.

Mutualisms are not acts of kindness: each partner pays costs and gains benefits, and a partner that stops delivering may be penalised. A simple way to model one is to let each species raise the other's carrying capacity. In the snippet below, each individual of one species adds 0.6 to the other's capacity, on top of a baseline capacity of 100.

```
# Mutualism: each species' carrying capacity rises with the partner's numbers. Invented
r, K, boost = 0.5, 100.0, 0.6
def run(partner_present, years=60):
    n1, n2 = 10.0, 10.0
    for t in range(years):
        k1 = K + boost * n2 if partner_present else K
        k2 = K + boost * n1 if partner_present else K
        n1 += r * n1 * (1 - n1 / k1)
        n2 += r * n2 * (1 - n2 / k2)
    return n1, n2
alone = run(False)
together = run(True)
print(f"alone:    {alone[0]:.0f} and {alone[1]:.0f}")
print(f"together: {together[0]:.0f} and {together[1]:.0f}")

# Output:
# alone:    100 and 100
# together: 250 and 250
```

Alone, each species settles at its own carrying capacity, 100. Together they settle at 250. That figure comes from a short piece of algebra: at the balance point, N = 100 + 0.6 × N, so N = 100 / (1 − 0.6) = 250. If the benefit were 1 or more, the model would let the two grow without limit. That cannot happen in nature, so real mutualisms are always capped by some other cost or shortage, which Exercise 2 explores.

Parasitism: Living on Another

A **parasite** feeds on a living **host** and harms it, but usually does not kill it quickly, since a dead host is a lost home. Ticks, fleas and lice live on the outside (**ectoparasites**), while tapeworms and many microbes live inside (**endoparasites**). Ticks are of practical interest in Scotland and the rest of the UK because they can carry Lyme disease. Some parasites do not live on the host at all: the cuckoo is a **brood parasite**, laying its egg in another bird's nest so that the host raises the chick.

Parasitism and predation share the same signs (+ / −), but the pattern differs. A predator kills one prey after another over a lifetime, while a parasite lives with one host for a long time. Parasites matter ecologically for three reasons: they can reduce the growth and breeding of a host, they can alter its behaviour, and they can tip the balance between competitors, as the disease carried by grey squirrels does for red squirrels in Chapter 7. A disease-causing parasite that spreads through a population is also a density-dependent brake of the kind described in Ecology and Ecosystems Fundamentals 6: the more crowded the hosts, the faster it spreads.

	PREDATION	MUTUALISM	PARASITISM
Effects	+ / −	+ / +	+ / −
Victim killed?	Yes, usually quickly	No: partners need each other	Rarely, and slowly
Typical numbers pattern	Linked cycles with a lag	Both rise or fall together	Parasite follows host numbers
Example	Lynx and hare	Tree and mycorrhizal fungus	Tick and deer

Common mistakes

Do not read the predator–prey cycle as predators "controlling" prey in a simple way: the two series drive each other, and food supply, disease and weather matter too. Do not assume that mutualism means friendship: it is a trade, and it can turn into exploitation if conditions change. Do not treat the signs (+, −, 0) as fixed labels, because an interaction can move between categories. Do not forget that in the basic cycle model the prey peak comes first and the predator peak after it, so a lag between the curves is normal. And remember that the snippets use invented numbers: they show how the model behaves, not how many foxes live on any real island.

Challenges

EXERCISE 1

Run the predator–prey model for 60 years with a time step of 0.01, starting from 40 prey and 20 predators, and storing every value in a list. Write a function that finds the years at which each series reaches a peak (a value higher than the one before and not lower than the one after). Print the prey peak years, the predator peak years, the average time between prey peaks, and how long after each prey peak the predators peak.

View solution

EXERCISE 2

Using the mutualism model with r = 0.5, K = 100 and starting numbers of 10 each, run 200 years for benefit values of 0.0, 0.2, 0.6 and 0.9. For each, print the final numbers and compare them with the formula K / (1 − benefit). What happens as the benefit approaches 1, and what would stop that in a real partnership?

View solution

EXERCISE 3

Run the predator–prey model for 120 years with the predator death rate d = 0.5 and then d = 0.7, and average the prey and predator numbers over the second half of each run. Compare your averages with the model's balance points (prey = d / c and predators = a / b). Which population changes when the predators' death rate rises, and which does not? Does that surprise you?

View solution

QUICK REFERENCE

- **Interactions by effect:** competition (−/−), predation and parasitism (+/−), mutualism (+/+), commensalism (+/0), amensalism (0/−)
- **Predator–prey cycle:** prey rise, predators follow, prey crash, predators fall; the predator peak lags the prey peak

- **Lotka–Volterra predator–prey:** prey change a×prey − b×prey×predators; predators change c×prey×predators − d×predators
- **Balance point:** prey = d / c and predators = a / b
- **Mutualism:** both partners gain; modelled by raising each other's carrying capacity, giving K / (1 − benefit)
- **Parasite:** lives on or in a host and harms it without usually killing it quickly; ectoparasites outside, endoparasites inside
- **Brood parasite:** a bird such as the cuckoo that leaves its eggs for another species to raise

COMING NEXT

Ecology and Ecosystems Fundamentals 9 asks what happens to a whole community after a fire, a flood or a clearing: how bare ground becomes woodland again, step by step.

Ecology and Ecosystems Fundamentals

Course 1 · Chapter 9 · Succession and Disturbance

Chapter 8 followed the numbers of species that eat, help and live on one another. Those interactions play out on a stage that is never still. Fires burn, rivers flood, storms fell trees and people clear land. This chapter asks what happens next. **Succession** is the gradual, fairly orderly change in the species of a community over time, and **disturbance** is the event that restarts it. You will meet the two main kinds of succession, the roles of early and late species, and the surprising idea that a moderate amount of disturbance can leave a place with more variety than none at all. As before, the maths is light and the ideas are tested in short Python simulations.

> **WHAT TO TRUST IN THIS CHAPTER**
>
> The Python snippets here and in the three exercise solutions were run, and the output in the comments is what Python printed. The rates, ages and thresholds are *invented round figures chosen for practice*, not measurements, and the random runs use a fixed seed so you should see the same numbers. The ecological background is written from general knowledge, not fetched from a source for this chapter, so check any real-world example you want to rely on, such as the island, volcano and woodland examples below, against a current textbook or a national agency.

Disturbance: The Reset Button

A **disturbance** is an event that removes living material or changes the environment enough to open up space and resources: a wildfire, a flood, a landslide, a storm, a grazing herd, a plough, a bulldozer. Ecologists describe a disturbance by how *big* it is, how *severe* (how much is destroyed), and how *often* it returns. Together these make up the **disturbance regime** of a place. A heath that burns every ten years and a woodland that burns once in several centuries have very different regimes, and different communities.

Disturbance is not always bad news. Many ecosystems depend on it: some pine cones open only after fire, river meadows are renewed by floods, and open grassland on a hillside may be kept open by grazing. What changes is the *pace* and *pattern* of recovery, which is succession.

Two Kinds of Succession

Succession starts from one of two very different situations.

- **Primary succession** begins on ground that has never held a community, or that has lost everything including the soil: bare rock exposed by a retreating glacier, fresh lava, a new sand dune, a new volcanic island. Soil has to be built first, slowly, from rock dust, dead organic matter and the work of the first colonisers.
- **Secondary succession** begins after a disturbance that leaves the soil in place, often with roots, seeds and buried organisms still alive: an abandoned field, a burned woodland, a clear-felled forest, a flooded meadow. It is usually much faster.

	PRIMARY SUCCESSION	SECONDARY SUCCESSION
Starting point	Bare rock, sand or new land with no soil	Existing soil, seeds and roots
First arrivals	Lichens, mosses and wind-blown seeds	Fast-growing weeds, grasses and sprouting stumps
Speed	Slow: often centuries	Faster: often decades
Example	Land uncovered by a melting glacier	A field left to grow wild
Common cause	Volcano, glacier, landslide scar	Fire, felling, flood, farming ending

Well-known cases include the land exposed as glaciers have retreated in Glacier Bay in Alaska, the island of Surtsey off Iceland, which appeared in the sea in the 1960s and has been followed by scientists ever since, and the slopes around Mount St Helens in the USA after the 1980 eruption, where some life survived under snow and in the soil. Closer to home, many Scottish uplands are a result of long human use: fire, grazing and felling hold back the woodland that would otherwise return, and fenced areas often show how quickly trees come back once grazing eases.

Who Arrives, and Why

The first species to arrive, the **pioneers**, are usually small, fast-growing and good at spreading seed widely, such as lichens, mosses, grasses, birch and willow. They cope with harsh, open, sunny conditions. Later species grow slower, live longer, and cope better in shade and in crowded ground, such as oak, beech and, in the Scottish Highlands, Scots pine. A community

that has settled into a long-lasting state is often called the **climax community**, although many ecologists now prefer to speak of a mature community that keeps shifting, because disturbance keeps interrupting it.

Why does one set of species replace another? Three explanations are usually discussed, and several can operate in the same place.

- **Facilitation:** early species improve conditions for later ones. Pioneers add organic matter to the soil, and some, like alders, host bacteria that add nitrogen (see Ecology and Ecosystems Fundamentals 4).
- **Inhibition:** early species make it harder for newcomers, for example by shading the ground or using up resources, so later species gain a place only when the early ones die or are damaged. This links to the competition in Ecology and Ecosystems Fundamentals 7.
- **Tolerance:** species simply differ in how they cope with limited resources. Those that manage with the least resource, such as shade-tolerant seedlings, eventually win.

Modelling Succession as Stages

A simple way to model succession is as a chain of stages: bare ground, grass, shrub, young woodland and mature woodland. Each year, a share of the land in one stage moves on to the next. The shares below are invented: bare ground moves on fast (30% a year), young woodland slowly (4% a year). The snippet starts with 100% bare ground and follows the percentages in each stage.

```
# Succession as a chain of stages: each year a share of each stage moves on to the next.
stages = ["bare ground", "grass", "shrub", "young wood", "mature wood"]
move = [0.30, 0.20, 0.10, 0.04]          # yearly share leaving bare, grass, shrub, youn
share = [1.0, 0.0, 0.0, 0.0, 0.0]        # start: 100% bare ground
print("year  " + "  ".join(f"{s:>11}" for s in stages))
for year in range(0, 101):
    if year % 20 == 0 or year in (5, 10):
        print(f"{year:4}  " + "  ".join(f"{x * 100:10.1f}%" for x in share))
    new = share[:]
    for i, m in enumerate(move):
        flow = share[i] * m
        new[i] -= flow
        new[i + 1] += flow
    share = new
```

```
# Output:
# year  bare ground        grass        shrub   young wood  mature wood
#    0       100.0%         0.0%         0.0%         0.0%         0.0%
#    5        16.8%        47.9%        31.0%         4.2%         0.1%
#   10         2.8%        23.7%        48.7%        22.5%         2.3%
#   20         0.1%         3.2%        29.8%        49.7%        17.2%
#   40         0.0%         0.0%         4.4%        39.6%        56.0%
#   60         0.0%         0.0%         0.5%        19.9%        79.6%
#   80         0.0%         0.0%         0.1%         9.1%        90.9%
#  100         0.0%         0.0%         0.0%         4.0%        95.9%
```

Read down the rows. The early stages rise and fall quickly: bare ground is almost gone within 10 years, and grass peaks and fades. Shrub is the largest stage at year 10, young woodland at year 20, and mature woodland keeps climbing, reaching 56% by year 40 and nearly 96% by year 100. This is the typical shape: each stage rises, peaks and gives way to the next, with the slowest step, young to mature woodland, setting the pace at the end. The model is a clear simplification, since real succession can stall, go sideways or vary from patch to patch, but it shows why you rarely find the same stage everywhere: the landscape holds a *mix* of stages at any moment. Exercise 1 compares starting from bare ground with starting from grass.

How Often Is Too Often?

Succession needs time. If disturbance returns before a mature community has formed, the land never gets there. Suppose a woodland patch counts as mature after 60 undisturbed years, and each year every patch has the same small chance, `p`, of being disturbed and reset to age zero. With many patches, you would expect a fixed share to be mature at any moment: the chance of a patch surviving 60 years undisturbed is $(1 - p)^{60}$. The simulation below checks that against the formula, using 2000 patches run for 1000 years.

```
import random
random.seed(1)
MATURE = 60                      # a patch counts as mature after 60 undisturbed years
def mature_share(p, patches=2000, years=1000):
    ages = [0] * patches
    for _ in range(years):
```

```
        ages = [0 if random.random() < p else age + 1 for age in ages]
    return sum(age >= MATURE for age in ages) / patches
print("disturbance/yr   mature (model)   mature (formula)")
for p in (0.001, 0.01, 0.02, 0.05, 0.10):
    print(f"{p:14.3f}   {mature_share(p) * 100:12.1f}%   {(1 - p) ** MATURE * 100:13.1f}%")

# Output:
# disturbance/yr   mature (model)   mature (formula)
#          0.001           94.0%           94.2%
#          0.010           55.3%           54.7%
#          0.020           29.5%           29.8%
#          0.050            4.5%            4.6%
#          0.100            0.3%            0.2%
```

The model and the formula agree to within about a percentage point, and the small gaps are random chance. The result is steep. A chance of 1 per cent a year (a disturbance every 100 years on average) leaves a little over half the patches mature. A chance of 5 per cent a year (every 20 years on average) leaves fewer than 5 in 100, and 10 per cent leaves almost none. A small change in how often a place is disturbed can therefore make a big change in what the landscape

The Intermediate Disturbance Idea

What disturbance does to variety is a famous question. With *very frequent* disturbance, only tough, quick-spreading pioneers can cope. With *very rare* disturbance, the strongest competitors take over and crowd the rest out, a result linked to the competitive exclusion principle in Ecology and Ecosystems Fundamentals 7. With *moderate* disturbance, patches of every age are present side by side, so pioneers, middle species and late species all find somewhere to live. This is the **intermediate disturbance hypothesis**. It has been supported in some studies and challenged in others, so treat it as a useful idea and not a law.

The snippet below uses the same patch model and sorts patches into three age bands: young (under 10 years), middle (10 to 39) and old (40 or more). A band counts as “present” if at least 5 per cent of the patches are in it, and the number of bands present stands in, very roughly, for the variety of habitats on offer.

```
import random
random.seed(2)
def groups_present(p, patches=2000, years=1000):
    ages = [0] * patches
    for _ in range(years):
        ages = [0 if random.random() < p else age + 1 for age in ages]
    bands = [sum(a < 10 for a in ages), sum(10 <= a < 40 for a in ages), sum(a >= 40 for
    return sum(b / patches >= 0.05 for b in bands), [round(b / patches * 100) for b in b
print("disturbance/yr   groups present   % of patches (young, middle, old)")
for p in (0.5, 0.1, 0.03, 0.01, 0.001):
    n, pct = groups_present(p)
    print(f"{p:14.3f}   {n:14}   {pct}")

# Output:
# disturbance/yr   groups present   % of patches (young, middle, old)
#          0.500                1   [100, 0, 0]
#          0.100                2   [64, 34, 2]
#          0.030                3   [25, 45, 30]
#          0.010                3   [9, 25, 66]
#          0.001                1   [1, 2, 97]
```

The pattern fits the idea. At 0.5 per year, nearly every patch is young. At 0.1 per year, young and middle patches exist but old ones are almost absent. At 0.03 and 0.01 per year, all three bands are common. At 0.001 per year, nearly all patches are old and only one band remains. Be careful about the leap from this toy to the real world: habitat variety is not the same as the number of species, and real species do not sort into three neat bands. The 5 per cent threshold is also arbitrary: Exercise 3 swaps it for a smoother measure.

	VERY FREQUENT DISTURBANCE	MODERATE DISTURBANCE	VERY RARE DISTURBANCE
Typical patch age	Mostly young	Mixed ages	Mostly old
Who does well	Pioneers only	Pioneers, middle and late species	Strong late competitors
Habitat variety	Low	High	Low

	VERY FREQUENT DISTURBANCE	MODERATE DISTURBANCE	VERY RARE DISTURBANCE
Example	A regularly ploughed field	A woodland with scattered storm gaps and a mosaic of ages	Old forest left untouched for centuries

Common mistakes

Do not picture succession as a fixed ladder that every place climbs to the same top: stages can be skipped, stalled or reversed, and the endpoint depends on climate, soil and chance. Do not call all disturbance damage, since many ecosystems and species depend on it. Do not treat “climax” as permanent: a mature community is still being disturbed and changing. Do not mix up primary and secondary succession: the difference is whether the soil survived, not how large the disturbance was. And remember that the snippets use invented numbers, which show how the model behaves rather than how long any real woodland takes to grow.

Challenges

EXERCISE 1

Using the stage model with the move rates 0.30, 0.20, 0.10 and 0.04, write a function that returns the first year in which mature woodland reaches 50% of the land and the first year in which bare ground falls below 1%. Run it from 100% bare ground (primary succession) and from 100% grass (a secondary succession where the soil is intact). Print both results and the difference. Why is the difference so small here, and what would you change in the model to match the real gap between the two kinds of succession?

View solution

EXERCISE 2

Use the patch model (mature after 60 years, 2000 patches, 1000 years, seed 1) to print the mature share for p = 0.001, 0.01, 0.02, 0.05 and 0.10 next to the formula $(1 - p)^{60}$. Then rearrange the formula to find the yearly disturbance chance at which only 10% of patches are mature, and check it with the model.

View solution

EXERCISE 3

Replace the “at least 5%” test with the Shannon diversity index, $H = -\sum s \times \ln(s)$, where s is the share of patches in each age band (young under 10, middle 10 to 39, old 40 or more). Calculate H for p = 0.5, 0.2, 0.1, 0.05, 0.03, 0.02, 0.01, 0.005 and 0.001 (seed 2), and find the disturbance rate with the highest index. Is the peak where the intermediate disturbance idea says it should be?

View solution

QUICK REFERENCE

- **Succession:** the gradual change in the species of a community over time; **disturbance:** an event that removes living material and restarts it
- **Disturbance regime:** the size, severity and frequency of disturbances in a place
- **Primary succession:** starts without soil (bare rock, new land); slow
- **Secondary succession:** starts with soil, seeds and roots intact (after fire, felling, farming); faster
- **Pioneers:** fast-spreading early species; **climax or mature community:** long-lived late species, still subject to disturbance
- **Facilitation, inhibition, tolerance:** three explanations of how early species give way to later ones
- **Share of mature patches:** $(1 - p)^{\text{age to maturity}}$ for yearly disturbance chance p
- **Intermediate disturbance hypothesis:** variety is highest at moderate disturbance; useful but debated

COMING NEXT

Ecology and Ecosystems Fundamentals 10 turns to biodiversity: how to measure the variety of life in a place, why it matters, and what the size and distance of an island can tell us about how many species it holds.

Ecology and Ecosystems Fundamentals

Course 1 · Chapter 10 · Biodiversity

Chapter 9 ended with a patchwork of young, middle-aged and old patches, and the idea that a mix of ages gives a place more variety. This chapter asks how to put a number on that variety. **Biodiversity**, short for biological diversity, is the variety of life at every level: the genes within a species, the species within a community, and the ecosystems within a landscape. You will meet the simple measures ecologists use, see why a plain species count can mislead, find out why bigger areas hold more species, and learn how the theory of island biogeography links a place's size and isolation to its number of species. As before, the maths is light and the ideas are tested in short Python simulations.

> **WHAT TO TRUST IN THIS CHAPTER**
>
> The Python snippets here and in the three exercise solutions were run, and the output in the comments is what Python printed. The counts, rates and areas are *invented round figures chosen for practice*, not measurements. No random numbers are used, so you should see exactly the same output. The ecological background is written from general knowledge, not fetched from a source for this chapter, so check any real-world example you want to rely on, such as the hotspot, island and nature-reserve examples below, against a current textbook or a national agency.

Three Levels of Variety

When people say “biodiversity”, they usually mean species, but ecologists count variety at three levels.

- **Genetic diversity:** the variety of genes within one species. A population with many gene variants is more likely to contain some individuals that cope with a new disease or a hotter summer. Crop varieties, such as the many kinds of apple or potato, are a familiar example.
- **Species diversity:** the number and balance of species in a community, which is what most of this chapter measures.
- **Ecosystem diversity:** the variety of habitats in a region, such as bog, heath, ancient woodland, estuary and sea cliff within a single stretch of Scottish coast.

Biodiversity is also unevenly spread. It tends to rise towards the tropics, and some areas, often called **biodiversity hotspots**, hold very large numbers of species found nowhere else. Chapter 5

showed how climate shapes the biomes (see Ecology and Ecosystems Fundamentals 5), and that is a large part of the reason: warm, wet, stable climates generally support more species than cold, dry or highly seasonal ones.

Measuring Species Diversity

The simplest measure is **species richness**: the number of species found. It is easy to understand, but it treats a species with one individual the same as a species with a thousand. Two sites can have the same richness and very different character. In one, the individuals are spread fairly evenly across the species. In the other, one species makes up nearly everything and the rest are rare. The spread is called **evenness**.

Indices that combine richness and evenness include two you will meet constantly. Let `s` be the share of all individuals that belong to one species.

- **Shannon index:** `H = -∑ s × ln(s)`. It is 0 when there is one species and rises as species are added or become more even. For `S` equally common species it equals `ln(S)`, the maximum possible.
- **Simpson index:** `1 - ∑ s²`. It is the chance that two individuals picked at random belong to *different* species. It is 0 for one species and approaches 1 for many even ones.
- **Evenness:** `H / ln(S)`, which runs from near 0 (one species dominates) to 1 (perfectly even).

The snippet below compares two invented meadow samples of 100 plants each. Both hold four species, so richness cannot tell them apart.

```
import math
communities = {
    "Meadow A": [25, 25, 25, 25],
    "Meadow B": [85, 5, 5, 5],
}
print("community   richness   Shannon H   Simpson 1-D   evenness")
for name, counts in communities.items():
    total = sum(counts)
    shares = [c / total for c in counts]
    h = -sum(s * math.log(s) for s in shares)
    simpson = 1 - sum(s * s for s in shares)
    print(f"{name:10}  {len(counts):8}   {h:9.2f}   {simpson:11.2f}   {h / math.log(len(
```

```
# Output:
# community   richness   Shannon H   Simpson 1-D   evenness
# Meadow A           4        1.39          0.75       1.00
# Meadow B           4        0.59          0.27       0.42
```

Both meadows have a richness of 4, but Meadow A has four species equally common (evenness exactly 1.00) while in Meadow B one species supplies 85 of the 100 plants. The Shannon index falls from 1.39 to 0.59 and the Simpson index from 0.75 to 0.27. If you walked across Meadow B you would see mostly one plant. The lesson is that richness alone hides how the individuals are shared out, so good surveys report richness *and* at least one index. Exercise 1 shows a case where the two disagree about which site is best.

	SPECIES RICHNESS	SHANNON INDEX	SIMPSON INDEX
Counts	Number of species	Species and their balance	Species and their balance
Most sensitive to	Rare species	Rare and common species alike	The commonest species
Range	1 upwards	0 upwards (maximum ln S)	0 to just under 1
Plain meaning	How many kinds?	How uncertain is the next individual?	Chance two picks differ

Measurement has a practical snag: you almost never count every individual, so the number of species you find depends on how hard you look. A bigger sample finds more species, especially rare ones. Ecologists therefore compare sites using the same effort, such as the same number of traps, the same plot size or the same minutes of birdsong survey, and they treat counts of rare species with care.

Why Biodiversity Matters

There are practical reasons and ethical ones, and both are worth stating plainly.

- **Ecosystem services:** pollination of crops, clean water, flood control from wetlands and floodplains, soil formation, and carbon storage in forests, peat and seas. The nutrient and water cycles of Ecology and Ecosystems Fundamentals 4 are carried out by living things.
- **Stability and resilience:** a community with many species, several of which do similar jobs, is more likely to keep working if one is lost, just as a team with several people able to cover a

task copes better when someone is off sick. This is often called the **insurance effect**. Experiments in grassland plots have often found that more diverse plots are more productive and steadier, though results depend on the system.

- **Resources and knowledge:** foods, fibres, timber and medicines, and genes that plant and animal breeders draw on. Many medicines began as natural compounds.
- **Intrinsic value:** many people feel that other species have a worth of their own, apart from any use to us. That is a values question, not one that data can settle, but it underlies much of conservation.

Bigger Areas, More Species

One of the oldest patterns in ecology is that larger areas hold more species. A square metre of grass holds a handful of plant species, a hectare holds many more, and a whole country holds thousands. The relationship usually follows a power law, the **species-area relationship**:

$S = c \times A^z$

Here S is the number of species, A the area, c a constant for the place and the group of organisms, and z the exponent. On a log-log graph this is a straight line with slope z. For islands, z values are often reported between about 0.2 and 0.35, and lower for samples taken from within one continuous habitat, though the number depends on the group and the region. The snippet uses invented values $c = 10$ and $z = 0.25$, and also asks what happens if area is cut to 10% of its size, for example by clearing or by building.

```
import math
c, z = 10, 0.25
print("area (km2)   species   loss if area cut to 10%")
for area in (1, 10, 100, 1000, 10000):
    species = c * area ** z
    left = c * (area * 0.1) ** z
    print(f"{area:10}   {species:7.1f}   {(1 - left / species) * 100:22.1f}%")
a1, a2 = 10, 10000
s1, s2 = c * a1 ** z, c * a2 ** z
print("fitted z from two points:", round(math.log(s2 / s1) / math.log(a2 / a1), 2))

# Output:
# area (km2)   species   loss if area cut to 10%
#          1      10.0                     43.8%
```

```
#         10      17.8                    43.8%
#        100      31.6                    43.8%
#       1000      56.2                    43.8%
#      10000     100.0                    43.8%
# fitted z from two points: 0.25
```

Every tenfold increase in area multiplies the species count by about 1.8 ($10^{0.25}$), so the same tenfold fall in area removes 43.8% of species at every size. This is why losing habitat hurts: a patch left after clearing eventually holds fewer species, even if the patch itself is untouched. The effect is not instant, since species can linger for years before dying out, a delay sometimes called **extinction debt**. A power law is a simplification, but it is a quick way to estimate the cost of shrinking a habitat. Exercise 2 fits `z` from data.

Island Biogeography

In the 1960s, the ecologists Robert MacArthur and Edward O. Wilson proposed a theory of **island biogeography** to explain why islands hold the numbers of species they do. It treats an island as a target for colonists from a mainland *species pool*, and says that the number of species on it settles at a balance between two opposing rates.

- **Immigration:** new species arriving. The rate falls as the island fills, because fewer of the species in the pool are still missing. It is higher for islands **near** the mainland and lower for **far** ones.
- **Extinction:** species on the island dying out. The rate rises as the island fills, because there are more species to lose and each has fewer individuals. It is higher on **small** islands and lower on **large** ones.

Where the two rates meet is the **equilibrium** number of species. Species keep coming and going, so the list changes over time (turnover) even though the total stays roughly steady. The theory predicts that large near islands hold the most species and small far islands the fewest, with the other two in between. It has been influential, and it also applies to habitat islands such as mountaintops, lakes, isolated woods and nature reserves surrounded by farmland. Real islands also differ in habitat variety, age and history, so the theory is a starting point, not a full explanation.

The model below is the simplest version. With a pool of 100 mainland species and `S` species present, the number gained each year is `i × (100 − S)` and the number lost is `e × S`. The numbers `i` and `e` are invented yearly rates: `i` is higher for a near island, and `e` is higher for a small one. Setting gains equal to losses gives the equilibrium `S = 100 × i / (i + e)`, which the simulation should approach.

```
POOL = 100
islands = {
    "large, near": (0.04, 0.02),
    "large, far":  (0.01, 0.02),
    "small, near": (0.04, 0.08),
    "small, far":  (0.01, 0.08),
}
print("island         year 10   year 50   year 200   formula")
for name, (i, e) in islands.items():
    s = 0.0
    row = {}
    for year in range(1, 201):
        s += i * (POOL - s) - e * s
        row[year] = s
    eq = POOL * i / (i + e)
    print(f"{name:12}   {row[10]:7.1f}   {row[50]:7.1f}   {row[200]:8.1f}   {eq:7.1f}")

# Output:
# island         year 10   year 50   year 200   formula
# large, near       30.8      63.6       66.7      66.7
# large, far         8.8      26.1       33.3      33.3
# small, near       24.0      33.3       33.3      33.3
# small, far         6.8      11.0       11.1      11.1
```

Each island climbs towards the value predicted by the formula and has settled there by year 200. The large near island holds 66.7 species and the small far island only 11.1. The small near island and the large far island both reach 33.3, which shows that nearness and size can offset each other. The path differs, though: after 10 years the small near island is well ahead of the large far one (24.0 against 8.8), because a near island is colonised quickly, yet in the end the two hold the same number. Exercise 3 asks how long each island takes to fill. Ideas like these inform conservation choices, such as whether to protect many small reserves or a few large ones, and

whether to link reserves with corridors. That so-called SLOSS debate ("single large or several small") has no single answer, since it depends on which species you most want to protect.

	LARGE, NEAR	LARGE, FAR	SMALL, NEAR	SMALL, FAR
Immigration	High	Low	High	Low
Extinction	Low	Low	High	High
Expected species	Most	Middle	Middle	Fewest
Real-world stand-in	A big wood near other woods	A big remote island	A small copse near a forest	A tiny isolated rock or hill-top

Biodiversity and Disturbance

This chapter links back to Ecology and Ecosystems Fundamentals 9. A landscape with disturbance at a moderate rate holds patches of every age, so more kinds of habitat are available, which is one route to higher diversity. It also links to Ecology and Ecosystems Fundamentals 7: where competition is strong and nothing interrupts it, a few species can take over, and diversity falls. Biodiversity is therefore not a fixed property of a place. It is the outcome of area, isolation, climate, history, competition and disturbance acting together.

Common mistakes

Do not equate biodiversity with the number of species alone: it also covers genes and ecosystems, and evenness matters. Do not compare species counts from samples of different sizes or effort. Do not assume more species is always better in every habitat: an open bog or a mountain top is naturally species-poor and valuable for exactly what lives there, and an invasive species can raise a count while lowering its worth. Do not treat the species-area law or the island model as exact predictions: both are simplified, and the figures in the snippets are invented to show how the models behave.

Challenges

EXERCISE 1

Three invented survey sites have these counts per species: Old wood [30, 22, 18, 12, 10, 8], Hay meadow [20, 18, 15, 14, 12, 11, 10] and Barley field [200, 10, 5, 3, 2, 1, 1, 1]. For each, calculate richness, the Shannon index, the Simpson index and evenness, then print the ranking of the sites by richness, by

Shannon and by Simpson. Which site changes position, and what does that say about using richness alone?

View solution

EXERCISE 2

Six invented islands have areas of 2, 5, 20, 60, 250 and 900 km² and 9, 12, 20, 27, 41 and 59 species. Fit `S = c × A^z` by taking logarithms of both columns and finding the least-squares line (no libraries needed). Print `z` and `c`, predict the species on islands of 5, 50 and 500 km², and say what percentage of species the fit predicts you would lose if an island's area were halved and if it were cut to a tenth.

View solution

EXERCISE 3

Use the island model with a pool of 100 species and the four islands from the chapter (large near: i 0.04, e 0.02; large far: i 0.01, e 0.02; small near: i 0.04, e 0.08; small far: i 0.01, e 0.08). For each island, print the equilibrium and the number of years the island takes to reach 90% of it. Compare the result with the rough formula 2.3 / (i + e). Which island fills fastest, and does that surprise you given that it has the fewest species?

View solution

QUICK REFERENCE

- **Biodiversity:** variety of life at three levels: genes, species and ecosystems
- **Species richness:** the number of species; **evenness:** how equally individuals are shared among them
- **Shannon index:** $H = -\sum s \times \ln(s)$, maximum $\ln(S)$; **Simpson index:** $1 - \sum s^2$, the chance two random individuals differ
- **Compare sites with equal sampling effort:** bigger samples find more species
- **Ecosystem services:** pollination, clean water, flood control, soil and carbon storage; **insurance effect:** many species protect against loss
- **Species-area relationship:** $S = c \times A^z$; cutting area to 10% with $z = 0.25$ removes about 44% of species

- **Island biogeography:** species number settles where immigration (higher when near) meets extinction (higher when small); equilibrium = pool × i / (i + e) in the simple model
- **Extinction debt:** species loss that lags behind habitat loss

COMING NEXT

Ecology and Ecosystems Fundamentals 11 turns to human impact: habitat loss, invasive species, pollution and climate change, and how the models in this chapter help us understand what is at stake.

Ecology and Ecosystems Fundamentals

Course 1 · Chapter 11 · Human Impact

Chapter 10 showed that the variety of life depends on area, isolation, climate and disturbance, and that a smaller area holds fewer species. This chapter turns to the biggest change in most of the world's ecosystems today: people. Humans are part of the biosphere, but we change it faster and more widely than any other single species. Ecologists often sort the main pressures into a memorable list, **HIPPO**: **H**abitat loss, **I**nvasive species, **P**ollution, human **P**opulation growth and **O**verharvesting. You will meet the first three in detail, plus climate change, which cuts across all of them, and see how the models from earlier chapters help us understand what is at stake. The maths stays light, and the ideas are tested in short Python simulations.

WHAT TO TRUST IN THIS CHAPTER

The Python snippets here and in the three exercise solutions were run, and the output in the comments is what Python printed. The areas, rates and concentrations are *invented round figures chosen for practice*, not measurements. No random numbers are used, so you should see exactly the same output. The ecological background is written from general knowledge, not fetched from a source for this chapter, so check any real-world example you want to rely on, such as the squirrel, bird-of-prey and mountain examples below, against a current textbook or a national agency.

Habitat Loss and Fragmentation

Most species die out because their home disappears. **Habitat loss** happens when woodland is cleared for farmland, wetlands are drained, or land is covered by buildings and roads. Its sister problem is **fragmentation**: what is left is broken into small patches separated by fields, roads or towns. Fragments suffer in three ways.

- **Less area:** by the species-area relationship of Ecology and Ecosystems Fundamentals 10, smaller patches hold fewer species.
- **More isolation:** as in the island model of Ecology and Ecosystems Fundamentals 10, a patch far from others is harder to recolonise after a species dies out in it.
- **More edge:** the edge of a patch has different light, wind and temperature from its middle, and is easier for predators and invasive species to enter. A small patch is nearly all edge.

The species-area power law gives a quick estimate of the cost. The snippet uses the same invented values as before, `c = 10` and `z = 0.25`, for a forest that starts at 1,000 km², and asks how many species the remaining forest could hold.

```
c, z = 10, 0.25
original = c * 1000 ** z
print("forest left   species kept   species lost")
for left in (100, 75, 50, 25, 10, 5):
    kept = c * (1000 * left / 100) ** z
    print(f"{left:10}%   {kept:11.1f}   {(1 - kept / original) * 100:11.1f}%")

# Output:
# forest left   species kept   species lost
#        100%         56.2           0.0%
#         75%         52.3           6.9%
#         50%         47.3          15.9%
#         25%         39.8          29.3%
#         10%         31.6          43.8%
#          5%         26.6          52.7%
```

Losing half the forest removes only about 16% of the species, which can sound mild, but by the time 90% is gone, 44% of the species are committed to disappear. Remember too the delay called extinction debt from Chapter 10: species often linger for years or decades after the clearing, so the full loss may not show up until long after the damage is done. Exercise 1 models that delay. Real fragments also lose more than area alone predicts, because of the edge effects above. A common remedy is to join patches with **corridors** such as hedgerows, river banks or green bridges over roads.

Invasive Species

A species moved by people into a place it never reached on its own is **introduced**. Most introduced species do little harm, and some are valuable crops. A minority spread, thrive and damage the native community, and these are called **invasive**. They are often free of the predators, parasites and diseases that kept them in check at home, so they can grow fast. They can compete with natives (Ecology and Ecosystems Fundamentals 7), eat them, carry disease or change the habitat itself.

Familiar UK examples include the grey squirrel, which has replaced the native red squirrel across most of England and Wales and carries a virus that harms reds but not greys, and Japanese knotweed, which grows through tarmac and shades out other plants. Around the world, rabbits in Australia, rats on remote islands and zebra mussels in North American lakes are classic cases. Islands are especially vulnerable, because their native species often evolved without ground predators or heavy competition.

The model below uses logistic growth from Ecology and Ecosystems Fundamentals 6. An invader starts at 10 individuals in a habitat that can hold `K = 1000`, with growth rate `r = 0.5` a year. Each year it grows, and then a fixed fraction is removed by control measures such as trapping.

```
K, r = 1000, 0.5
def run(start, removal, years):
    n = start
    out = {}
    for year in range(1, years + 1):
        n = n + r * n * (1 - n / K)
        n = n * (1 - removal)
        out[year] = n
    return out
print("year   no control   30% removed yearly   50% removed yearly")
a, b, c = run(10, 0, 30), run(10, 0.3, 30), run(10, 0.5, 30)
for y in (5, 10, 15, 20, 30):
    print(f"{y:4}   {a[y]:10.0f}   {b[y]:18.0f}   {c[y]:18.0f}")

# Output:
# year   no control   30% removed yearly   50% removed yearly
#    5           73                   13                    2
#   10          411                   16                    1
#   15          919                   19                    0
#   20          997                   24                    0
#   30         1000                   35                    0
```

Uncontrolled, the invader is slow at first and then explodes: from 73 to 411 between years 5 and 10, and it has filled the habitat by year 20. With 30% removed each year it is held far down but still creeping upwards, and with 50% removed it is wiped out. Two lessons follow. First, **early action is far cheaper**: control is easiest when numbers are small, before the steep part of the growth curve. Second, **effort must beat the growth rate**: a removal that is slightly too low

never wins. Prevention, such as checking boats, plants and ballast water, is cheaper still. Exercise 2 finds the threshold.

Pollution

Pollution is any substance or energy added to an environment faster than it can be broken down or absorbed. Ecologists group it by where it goes and what it does.

- **Nutrient pollution:** fertiliser and sewage washing into rivers and lakes add nitrogen and phosphorus (Ecology and Ecosystems Fundamentals 4). Algae boom, then die and rot, using up the oxygen that fish need. This is called **eutrophication**.
- **Toxic chemicals:** pesticides, metals and industrial compounds can poison wildlife directly.
- **Plastics:** break into smaller and smaller pieces, which are swallowed by animals and now turn up in soils, rivers and the deep sea.
- **Air pollution:** gases such as sulphur dioxide produced *acid rain* that damaged lakes and forests in the past, and nitrogen compounds still add unwanted fertiliser to heaths and bogs.
- **Noise and light:** traffic noise can drown out bird song, and street lighting disturbs insects, bats and migrating birds.

Some pollutants are especially worrying because of the trophic structure from Ecology and Ecosystems Fundamentals 2 and 3. A substance that is fat-soluble and slow to break down is stored in the body, so each animal carries the load of everything it has eaten. This build-up in one organism is **bioaccumulation**, and its increase up the food chain is **biomagnification**. Because only about a tenth of the energy passes to the next level, a predator needs many prey, and the pollutant in all of them ends up in one body. This is why persistent pesticides such as DDT harmed birds of prey, which sit at the top of the web, even when the water they drained into looked clean. The snippet multiplies an invented concentration tenfold at each step.

```
levels = ["water", "algae", "small fish", "large fish", "bird of prey"]
factor = 10
conc = 0.0002
print("level           concentration (mg/kg)")
for i, name in enumerate(levels):
    print(f"{name:14}  {conc * factor ** i:18.4f}")

# Output:
```

```
# level           concentration (mg/kg)
# water                        0.0002
# algae                        0.0020
# small fish                   0.0200
# large fish                   0.2000
# bird of prey                 2.0000
```

A level in the water that is a tiny 0.0002 mg/kg becomes 2.0 mg/kg at the top, ten thousand times higher. Real factors differ widely between substances and food webs, and a tenfold step is only a teaching figure. The practical result is that regulators monitor top predators, such as otters, seals and eagles, as warning signs, and that banning the worst chemicals has let several bird populations recover.

Climate Change

Burning coal, oil and gas, clearing forests and farming release carbon dioxide, methane and other **greenhouse gases**. Chapter 4 followed carbon through the biosphere (see Ecology and Ecosystems Fundamentals 4): fossil fuels are carbon that spent millions of years out of the cycle, and burning them returns it to the atmosphere within decades. More greenhouse gas traps more heat, and the global average temperature has been rising. Ecosystems respond in several ways.

- **Range shifts:** many species move towards the poles or up hillsides to stay in the climate they suit. Some birds, butterflies and marine fish have been recorded moving in this direction.
- **Timing mismatches:** spring is arriving earlier in many places. If a caterpillar peak moves earlier but the chicks that feed on it still hatch on the old schedule, there is a gap between supply and demand. Such changes in the timing of life events are called **phenology** shifts.
- **Ocean change:** the sea absorbs some of the extra carbon dioxide, which makes it slightly more acidic and harder for shell-building animals and corals to form their skeletons. Warmer water also causes coral bleaching.
- **Biome shifts and disturbance:** a warmer climate changes which biomes (Ecology and Ecosystems Fundamentals 5) a region can support, and can increase the frequency of fire, drought and storm, the disturbances of Ecology and Ecosystems Fundamentals 9.

Species that live on mountains and in cold regions face a particular squeeze, because moving upwards or polewards eventually runs out of land. Exercise 3 models a species pushed up a

mountain.

	HABITAT LOSS	INVASIVE SPECIES	POLLUTION	CLIMATE CHANGE
Main effect	Less area, more isolation	Competition, predation, disease	Poisoning, oxygen loss	Shifted ranges and timing
Chapter model	Species-area law, island model	Logistic growth, competition	Nutrient cycles, food chains	Biomes, disturbance
Typical example	Hedgerow or wood cleared for building	Grey squirrel, Japanese knotweed	Fertiliser run-off, persistent pesticides	Earlier spring, coral bleaching
Common remedy	Protect habitat, add corridors	Prevent, detect early, control	Reduce at source, ban the worst	Cut emissions, protect refuges

Pressures Rarely Act Alone

The four pressures interact. A fragmented habitat is easier for invaders to enter. A species already weakened by pollution copes less well with a heat wave. Climate change lets some invaders spread to places that used to be too cold for them. Ecologists therefore talk about **cumulative impacts** and about **resilience**: a diverse, connected ecosystem (Ecology and Ecosystems Fundamentals 10) has a better chance of absorbing shocks than a small, isolated one. The same chain of thinking gives a hopeful point. Because the pressures are caused by human choices, they can also be reduced by human choices, which is the subject of the final chapter.

Common mistakes

Do not treat every introduced species as invasive: most are harmless, and the harmful ones should be judged by their effects. Do not assume that a lost species disappears the moment its habitat does, because extinction debt means the loss may come later, and equally do not assume that the lost species can always come back. Do not read a model's figures as predictions for a real place: the numbers in the snippets are invented to show how the models behave. Finally, do not blame one cause when there are several, since habitat loss, invasives, pollution and climate usually act together.

Challenges

EXERCISE 1

A forest of 1,000 km^2 is cleared by 5% of what is left each year for 20 years, then left alone for 10 more. The species-area law with `c = 10` and `z = 0.25` gives the number the remaining forest can hold. Each year, the species actually present drift 10% of the way down towards that number. Print the forest area, the number it can hold, the number still present and the extinction debt (present minus can-hold) at years 5, 10, 20 and 30, then the total committed loss and the loss actually suffered by year 30.

View solution

EXERCISE 2

Using the invader model from the chapter (`K = 1000`, `r = 0.5`, a starting population of 10), write a function that returns the population after 60 years for a given yearly removal fraction. Print the result for 0%, 10%, 20%, 30%, 33%, 34%, 40% and 50%, then find the smallest whole-percent removal that makes the population fall below its starting 10. Compare the answer with the rule of thumb `r / (1 + r)`. Why does it matter that the removal is repeated every year?

View solution

EXERCISE 3

A mountain is 1,000 m high. The temperature at its foot is 12 °C and falls by 6.5 °C for every kilometre of height. A plant lives only where the mean temperature is between 8 and 12 °C. For warming of 0 to 8 °C in whole degrees, print the lowest and highest height at which the plant can live (keeping to the mountain) and the length of the band. Then find the first warming, in steps of 0.5 °C, at which no suitable height is left. What does this say about mountain-top species?

View solution

QUICK REFERENCE

- **HIPPO:** habitat loss, invasive species, pollution, human population, overharvesting
- **Habitat loss and fragmentation:** less area, more isolation, more edge; the species-area law estimates the cost (10% of the forest left with z = 0.25 means about 44% of species lost)
- **Extinction debt:** species loss that lags behind habitat loss; **corridors** connect fragments

- **Invasive species:** introduced species that spread and harm natives; act early, because removal must beat the growth rate (about r / (1 + r) a year in the simple model)
- **Pollution:** nutrient (eutrophication), toxic, plastic, air, noise and light; **biomagnification** raises persistent toxins up the food chain
- **Climate change:** range shifts, timing mismatches (phenology), ocean acidification and more frequent disturbance; mountain and polar species can run out of room
- **Cumulative impacts:** the pressures interact, and diverse, connected ecosystems are more resilient

COMING NEXT

Ecology and Ecosystems Fundamentals 12 is the final chapter, on conservation and restoration: protected areas, rewilding, and practical ecology you can do yourself.

Ecology and Ecosystems Fundamentals

Course 1 · Chapter 12 · Conservation and Restoration

Chapter 11 ended with a hopeful point: pressures caused by human choices can be reduced by human choices. This final chapter is about those choices. **Conservation** is the protection of species and ecosystems, and **restoration** is the repair of ones that have been damaged. Both use everything in this course: area and isolation from the species-area relationship, growth and carrying capacity from population models, succession for recovery, and diversity measures to check whether it worked. The maths stays light, and the ideas are tested in short Python simulations. The chapter closes with practical ecology you can do yourself.

> **WHAT TO TRUST IN THIS CHAPTER**
>
> The Python snippets here and in the three exercise solutions were run, and the output in the comments is what Python printed. The areas, rates and survey counts are *invented round figures chosen for practice*, not measurements. No random numbers are used, so you should see exactly the same output. The ecological background is written from general knowledge, not fetched from a source for this chapter, so check any real-world example you want to rely on, such as the reintroduction, reserve and target examples below, against a current textbook or a national agency.

Protected Areas

The oldest tool is to set land or sea aside. **Protected areas** range from strict nature reserves, where almost nothing is allowed except study, to national parks and marine protected areas that combine wildlife protection with recreation, to working landscapes where farming continues under agreed rules. In the UK, national parks, sites of special scientific interest and local nature reserves are all examples, and many other countries have comparable systems. International agreements have set a target of protecting about 30% of land and sea by 2030, but ecologists point out that the number matters less than *where* the protected land is and *how well* it is managed. A reserve that exists only on a map, with no staff or enforcement, is sometimes called a **paper park**.

Good design draws on Ecology and Ecosystems Fundamentals 10. Bigger areas hold more species and are less exposed to edge effects. Areas connected by corridors can exchange individuals. Areas that cover a range of habitats and heights give species somewhere to go as the climate changes. A long-running debate known as **SLOSS** (single large or several small) asks

whether one big reserve beats several small ones of the same total area. The answer depends on how much the small ones overlap in species. The snippet uses the species-area law with the invented values `c = 10` and `z = 0.25`, and assumes that each extra small reserve shares a fixed fraction of its species with the others.

```
c, z = 10, 0.25
print("overlap   one reserve of 100   four reserves of 25")
for overlap in (0.5, 0.7, 0.9):
    one = c * 100 ** z
    each = c * 25 ** z
    four = each * (1 + 3 * (1 - overlap))
    print(f"{overlap:7.1f}   {one:18.1f}   {four:19.1f}")

# Output:
# overlap   one reserve of 100   four reserves of 25
#     0.5                31.6                  55.9
#     0.7                31.6                  42.5
#     0.9                31.6                  29.1
```

If the small reserves hold very different species (overlap 0.5), four of them protect far more than one large one: 55.9 species against 31.6. If they hold mostly the same ones (overlap 0.9), the single large reserve wins narrowly, 31.6 against 29.1. Real planners therefore ask which species are involved: large reserves suit wide-ranging animals and species that cannot survive near an edge, while several small, varied sites spread the risk of a single fire, flood or disease. Most designs aim for a network of reserves of different sizes joined by habitat in between.

Species Recovery and Reintroduction

Protecting land is not enough when a species is down to a handful of animals. Recovery programmes can include captive breeding, **translocation** (moving animals to a new place) and **reintroduction** (returning a species to somewhere it used to live). Reintroductions are often cited in the UK and elsewhere, for example the red kite in Wales and England, the white-tailed eagle in Scotland and the beaver in Scottish and English rivers, while the wolf in Yellowstone National Park is the textbook example of a trophic cascade (Ecology and Ecosystems Fundamentals 3). Reintroductions work best when the original cause of the loss has been dealt with, when enough animals are released, and when local people are involved.

The size of the founding group matters because of the population models of Ecology and Ecosystems Fundamentals 6. Here logistic growth uses `r = 0.4` and `K = 300`, but a fixed number of animals is also lost every year to causes such as road deaths or illegal killing. The snippet follows three cases.

```
r, K = 0.4, 300

def herd(start, loss, years=40):
    n = start
    out = [n]
    for _ in range(years):
        n = n + r * n * (1 - n / K) - loss
        n = max(n, 0)
        out.append(n)
    return out

a, b, d = herd(8, 0), herd(8, 3), herd(20, 3)
print("year   start 8, no loss   start 8, 3 lost a year   start 20, 3 lost a year")
for y in (0, 5, 10, 20, 30, 40):
    print(f"{y:4}   {a[y]:16.0f}   {b[y]:22.0f}   {d[y]:23.0f}")

# Output:
# year   start 8, no loss   start 8, 3 lost a year   start 20, 3 lost a year
#    0                8                        8                         20
#    5               40                        9                         62
#   10              147                       15                        176
#   20              297                      132                        290
#   30              300                      288                        292
#   40              300                      292                        292
```

Without losses the herd of 8 reaches almost 300 within 30 years. With three animals lost every year the same herd stays tiny for a decade (9 at year 5, 15 at year 10) before it finally takes off, while a starting group of 20 gets there much sooner. A fixed loss creates a **turning point**: below it, growth cannot keep up with the loss and the population dies out. Exercise 1 finds it. The practical message is to release enough individuals, to remove the causes of loss, and to expect that small populations are fragile and can be tipped to extinction by chance events.

Rewilding and Restoration

Restoration ecology repairs a damaged ecosystem. The idea ties directly to Ecology and Ecosystems Fundamentals 9: if the soil, seed supply and nearby habitat are still there, **succession** will do much of the work on its own, and the manager's job is to remove the cause of damage, then wait. Where those are missing, people help, by planting trees, re-wetting drained peat, removing dams, re-profiling straightened rivers or reseeding meadows. Restoration has several flavours.

- **Passive restoration:** stop the damage, such as grazing or draining, and let nature recover. It is cheap but can be slow.
- **Active restoration:** plant, sow, re-shape land or return water. It costs more and is faster.
- **Rewilding:** restore natural processes such as grazing, flooding and predation, often by bringing back *keystone species* (Ecology and Ecosystems Fundamentals 3), so that the ecosystem manages itself with less human control. Supporters point to richer habitats; critics point to costs, farming concerns and the safety of people and livestock, so local agreement is central.
- **Nature-based solutions:** work with ecosystems to solve human problems, such as restoring saltmarsh to soften storm surges, or peatland and woodland to store carbon (Ecology and Ecosystems Fundamentals 4).

A restored wood does not appear overnight, and a simple model shows why. The cover of woodland on a site grows by *natural spread* from seed sources, which is logistic like the populations of Ecology and Ecosystems Fundamentals 6, plus a fixed amount of planting each year. Exercise 2 makes this concrete. A slow start from low cover is the main reason restoration targets are measured in decades.

Restoration is a gamble if it ignores the lessons of the earlier chapters. Planting a single fast-growing species gives trees but not a woodland community. A restored site has to be judged not by what was put in but by what has *come back*, and a lasting community of native species is a better sign than a planting count. This is why monitoring (see below) is part of every serious project.

	PROTECT	RECOVER SPECIES	RESTORE	REWILD
Goal	Stop loss in what remains	Bring a species back from the edge	Repair a damaged habitat	Return natural processes
Chapter model	Species-area law, island model	Logistic growth, small populations	Succession, disturbance	Food webs, keystone species

	PROTECT	RECOVER SPECIES	RESTORE	REWILD
Typical example	Nature reserve, marine protected area	Captive breeding, reintroduction	Re-wetting peat, planting native woodland	Beavers, free-roaming grazers
Main risk	Paper parks, isolation	Too few animals, causes of loss remain	Slow results, wrong species	Local opposition, unintended effects

Practical Ecology You Can Do

You do not need a degree to help. The same ideas scale down to a garden, a school ground, a balcony or a local park, and ecologists rely heavily on volunteers.

- **Add habitat:** plant native flowers and shrubs, leave a patch of long grass, build a pond, and put up nest boxes and bug hotels. Even a small patch adds area, and a row of gardens with a gap in each fence for hedgehogs can act as a corridor.
- **Reduce pressures:** avoid pesticides and peat-based compost, switch off unneeded outdoor lights, and do not release pets or dump garden plants into the wild, which prevents new invasive species (Ecology and Ecosystems Fundamentals 11).
- **Join citizen science:** garden bird counts, butterfly transects, pond surveys and bioblitzes, in which volunteers record as many species as they can in a day, feed national databases. Phone apps can help identify plants and birds.
- **Survey your own patch:** use the methods of Ecology and Ecosystems Fundamentals 10. Count in the same way, on the same route, in the same season, so that the numbers can be compared fairly, and write down effort as well as results.
- **Help locally:** volunteer with a nature reserve or river-cleaning group, support local wildlife trusts, and speak up in planning decisions that affect nearby habitats.

Monitoring is how an ecologist tells whether any action worked. The snippet below compares two invented garden bird counts, before and after the changes, using the Shannon index and evenness of Ecology and Ecosystems Fundamentals 10. The total numbers are almost the same, but the community is not.

```
import math
```

```
def shannon(counts):
    total = sum(counts.values())
    return -sum(n / total * math.log(n / total) for n in counts.values())

lawn = {"blackbird": 12, "starling": 25, "house sparrow": 9, "wood pigeon": 4}
garden = {"blackbird": 10, "starling": 12, "house sparrow": 9, "wood pigeon": 4,
          "blue tit": 8, "robin": 6, "wren": 3}
print("survey    species   birds      H   evenness")
for name, counts in (("lawn", lawn), ("garden", garden)):
    h = shannon(counts)
    s = len(counts)
    print(f"{name:8} {s:8} {sum(counts.values()):7} {h:6.2f} {h / math.log(s):10.2f}")

# Output:
# survey    species   birds      H   evenness
# lawn            4      50   1.20       0.87
# garden          7      52   1.86       0.95
```

The garden gained three species and, just as important, became more even: the starling no longer dominates, so H rose from 1.20 to 1.86. The raw bird count (50 and 52) would have

which is what turns a hobby into data. Exercise 3 asks you to repeat the analysis.

> Common mistakes
> Do not judge a reserve by its size on the map without asking how it is managed and connected. Do not release animals without removing what caused the loss, because they will simply die out again. Do not count trees planted as if that were the same as a restored woodland: judge the community that returns. Do not read a model's figures as predictions for a real place, because the numbers in the snippets are invented to show how the models behave. Finally, do not compare surveys made in different seasons or with different effort, since the difference may come from the method and not from the ecosystem.

Challenges

EXERCISE 1

Use the herd model from the chapter (`r = 0.4`, `K = 300`, with 3 animals lost every year after growth, and the population never below zero). For starting herds of 6, 7, 8, 12, 20 and 40, print the first year in

which the herd reaches 150, or say that it never does within 60 years. Then find the smallest starting herd that ever reaches 150, and compare it with the turning point where growth just equals the loss. Why is a larger founding group safer?

View solution

EXERCISE 2

Model woodland cover as a fraction of the site: each year it rises by a fixed amount of planting plus `0.05 * cover * (1 - cover)` of natural spread, starting at 5%. Print the number of years to reach 50% cover for planting rates of 0%, 1%, 2%, 4% and 8% a year. Then, with no planting at all, print the years needed from starting covers of 2%, 5%, 10% and 20%. What does this say about starting restoration where some woodland already survives?

View solution

EXERCISE 3

A lawn is surveyed in year 0 (blackbird 12, starling 25, house sparrow 9, wood pigeon 4) and the same garden in year 3 after planting shrubs and adding a pond (blackbird 10, starling 12, house sparrow 9, wood pigeon 4, blue tit 8, robin 6, wren 3). Write a Shannon index function and print the number of species, the number of birds, H and evenness for each survey, then the change in H and the species gained. Which is the stronger sign of change, the bird count or the index, and why?

View solution

QUICK REFERENCE

- **Protected areas:** reserves, parks and marine areas; design uses size, connection and variety, and management matters more than the map
- **SLOSS:** one large or several small; several small win when they hold different species, one large wins when they overlap heavily
- **Reintroduction:** remove the cause of loss, release enough animals, involve local people; a fixed yearly loss creates a turning point below which a population dies out
- **Restoration:** passive (stop and wait), active (plant and reshape), rewilding (restore processes), nature-based solutions; succession does much of the work if soil and seeds remain

- **Monitoring:** repeat the same method in the same season; diversity indices show changes that raw counts hide
- **What you can do:** add habitat, reduce pressures, join citizen science, survey your patch, help locally

Course Complete

You have travelled from a single organism to the whole biosphere and back again to a garden pond. The main threads of the course were these.

- **Structure:** life is organised in levels, from organism to biosphere (Ecology and Ecosystems Fundamentals 1).
- **Energy and matter:** energy flows one way through food chains and webs, while nutrients cycle (Ecology and Ecosystems Fundamentals 2 to 4).
- **Place:** climate shapes the biomes of the world (Ecology and Ecosystems Fundamentals 5).
- **Interactions:** populations grow, compete, hunt, cooperate and recover from disturbance (Ecology and Ecosystems Fundamentals 6 to 9).
- **Diversity and people:** biodiversity can be measured, and human pressures can be understood, reduced and in part reversed (Ecology and Ecosystems Fundamentals 10 to 12).

The most useful habit to keep is the ecologist's question: what is the pattern, what is causing it, and how would I tell? You now have the vocabulary and a toolkit of simple models to ask it with. A natural next step is a course on a related science or on a specific habitat or region, or just a notebook, a pair of binoculars and a patch of ground you visit regularly.