



OFFICE & PRODUCTIVITY SOFTWARE

Word & PowerPoint

Styles & Document Structure · Page Layout & Section Breaks

Tables of Contents, Captions & Mail Merge

Track Changes & Real Collaboration

The Slide Master & Effective Slide Design

Presenting, Exporting & a Combined Capstone Project

Philip Osztromok

Generated with Claude

Table of Contents

Word & PowerPoint — 12 Chapters

CHAPTER	TITLE
Chapter 1	Two Different Mental Models: A Flowing Document vs. a Deck of Discrete Slides
Chapter 2	Styles and the Style Gallery: Why “Just Bold It” Is the Wrong Habit
Chapter 3	Page Layout: Margins, Page Breaks vs. Section Breaks
Chapter 4	Automatic Tables of Contents, Captions & Cross-References
Chapter 5	Headers, Footers & Page Numbers in Long Documents
Chapter 6	Track Changes and Comments: Real Collaboration in Word
Chapter 7	Mail Merge: Generating Many Personalized Documents from One Template
Chapter 8	Slide Layouts and the Slide Master: Consistency Without Repetitive Work
Chapter 9	Effective Slide Design: Content Density, Visual Hierarchy, and “Death by Bullet Points”
Chapter 10	Animations and Transitions, Used Sparingly
Chapter 11	Presenting: Presenter View, Rehearse Timings, and Exporting
Chapter 12	Capstone: A Business Report in Word and Its Companion Deck in PowerPoint

Word & PowerPoint

Chapter 1 · Two Different Mental Models: A Flowing Document vs. a Deck of Discrete Slides

The Excel course opened with a single idea that everything else built on: Excel is a grid of independent, addressable cells. Word and PowerPoint share the same Ribbon, the same theme system, even the same file-saving habits — but neither one works anything like a grid. Get each app's own real mental model straight in this chapter, and every later chapter's tools will feel like natural extensions of something you already understand, exactly the role Chapter 1 played in the Excel course.

The Three Content Models of Microsoft Office

One suite, one shared interface, and three genuinely different ways of organizing content underneath it:

APP	BASIC UNIT OF CONTENT	HOW CONTENT IS ORGANIZED
Excel	A cell	A fixed grid — every cell has its own independent address
Word	A continuous stream of text	One flowing document — content has no fixed position, it reflows as you edit
PowerPoint	A slide	A fixed, ordered sequence of independent slides — each its own separate canvas

What Word Actually Is: A Flowing Document

Word has no addressable "cells" at all. Text simply **flows** — type a new sentence on page 3, and everything after it shifts down to make room, potentially pushing content onto entirely new pages, with no manual intervention. There's no equivalent of an Excel cell address you could point a formula at; a Word document's content is defined purely by its position in one continuous stream, from the very first character to the very last.

This is exactly why "the third paragraph" or "the heading on page 4" are meaningless as fixed addresses in Word the way B7 is fixed in Excel — add one sentence earlier in the document, and what used to be on page 4 might now be on page 5, with nothing in the document's own structure needing to be manually updated to make that happen.

What PowerPoint Actually Is: A Deck of Discrete, Independent Slides

PowerPoint sits at the opposite extreme. A presentation is a fixed, ordered sequence of **slides**, and each slide is its own separate, independent canvas — editing slide 5 has zero effect on the layout, content, or size of slide 4 or slide 6. Nothing "flows" between slides the way text flows continuously through a Word document.

This makes a slide the closest thing PowerPoint has to Excel's own independent, addressable unit — just at a far coarser granularity. A cell holds one value; a slide holds an entire arrangement of text, images, and shapes, but the underlying principle is the same: each one is independently addressable, and editing one doesn't reshuffle its neighbours' own internal content the way editing text in Word does.

The Shared Interface: Ribbon, Themes, and Consistency Across the Suite

Despite these genuinely different content models, all three apps present themselves through the same **Ribbon** — tabbed groups of related commands, consistently laid out across Word, Excel, and PowerPoint. More usefully, a custom **Theme** (a saved combination of colours, fonts, and effects, accessible from each app's own Design tab) can be applied consistently across a Word report, its companion PowerPoint deck, and even an Excel workbook's own chart styling — genuinely useful the moment a report and its summary slide deck need to visually match without manually re-picking the same colours and fonts in two or three separate places.

Navigating Each App's Own Space

Each content model comes with its own natural way of moving around:

- **Word** — scrolls continuously through the flowing document; **Ctrl+Home** jumps to the very start, **Ctrl+End** to the very end — the same two shortcuts Excel Fundamentals Chapter 1 introduced for jumping to a worksheet's own edges, doing the equivalent job here.
- **PowerPoint** — Normal view's slide thumbnail pane on the left lets you jump directly to any discrete slide; **Slide Sorter** view zooms out to show every slide as a small thumbnail at once, letting you drag whole slides into a new order the way you might rearrange playing cards — something Word has no real equivalent of, since Word's content has no discrete units to physically drag around in the first place.

THIS CHAPTER'S CONTRAST PREVIEWS THE WHOLE COURSE

Chapter 2's Word Styles and Chapter 8's PowerPoint Slide Master both solve the exact same underlying problem — "define formatting once, apply it everywhere, and update it in one place later" — but each has to solve it in a way that fits its own app's content model: Styles work on flowing paragraphs and headings; a Slide Master works on a fixed layout applied to a whole slide at once. Recognising that shared goal, expressed two different ways, will make both chapters land faster than treating them as unrelated features to memorise separately.

ALIGNING CONTENT WITH SPACES OR TABS BREAKS THE MOMENT ANYTHING UPSTREAM CHANGES

A habit that feels natural coming from Excel's own grid — lining things up by eye with the spacebar, the way you might nudge a value into a specific column — actively fights Word's flowing text model. Because text reflows continuously, manually-typed spaces used for alignment shift out of place the instant a font size changes, text is added earlier in the line, or the document is opened on a different device with different default fonts. Word's own **Tables** and **Tab Stops** exist specifically to create genuine, stable alignment that survives exactly this kind of change — reaching for the spacebar to align anything beyond a single word is almost always the wrong tool.

Hands-On Exercises

EXERCISE 1

Open a blank Word document and type several paragraphs of placeholder text (enough to fill more than one page). Insert a new sentence at the very beginning of the document. Describe exactly what happens to every paragraph that came after it, and explain why nothing needed to be manually adjusted for this to happen.

 [View solution](#)

EXERCISE 2

Open a blank PowerPoint presentation with 5 slides, each containing different placeholder text. Switch to Slide Sorter view and drag slide 5 to become slide 2. Explain what happened to the actual content of every other slide, and why this specific operation has no real equivalent in Word.

 [View solution](#)

EXERCISE 3

A colleague built a simple two-column list in Word by typing an item, pressing Space repeatedly to line up a second column of numbers, then pressing Enter, repeating this for every row. After changing the document's font, every row's numbers no longer line up. Explain exactly why this happened and what feature they should have used instead.

 [View solution](#)

CHAPTER 1 QUICK REFERENCE

- **Excel:** a grid of independently addressable cells
- **Word:** one continuous flowing stream of text — no fixed addresses, content reflows automatically
- **PowerPoint:** a fixed, ordered sequence of independent slides — the closest thing to Excel's own "independent unit," just at a coarser grain
- **Ctrl+Home / Ctrl+End** — jump to the start/end of a Word document, the same shortcuts Excel uses for its own worksheet edges
- **Slide Sorter view** — rearrange whole slides like physical cards; Word has no real equivalent, since it has no discrete units to drag
- A shared **Theme** can keep a Word report and its PowerPoint companion deck visually consistent
- Use **Tables** or **Tab Stops** for real alignment in Word — spacebar alignment breaks the moment anything upstream changes

Word & PowerPoint

Chapter 2 · Styles and the Style Gallery: Why "Just Bold It" Is the Wrong Habit

Selecting text and clicking Bold feels like the obvious way to make a heading stand out — and visually, it works. But it leaves Word with no idea that text was ever *meant* to be a heading, only that it happens to look bold and slightly bigger. Styles solve this properly, the exact same way Excel Fundamentals Chapter 6's Cell Styles did for spreadsheets: name a formatting definition once, apply it by name, and update every instance centrally later.

Direct Formatting vs. Styles: What's Actually Different

Selecting text and clicking Bold, changing its font, or bumping its size is **direct formatting** — it changes exactly the selected text and nothing else, with no name or meaning attached to what you did. A **Style** (Home tab → Styles gallery) instead applies a **named, reusable definition** — apply "Heading 1" to three different paragraphs throughout a document, and later changing the Heading 1 style's own definition updates all three at once, instantly, with no need to reselect any of them.

Paragraph Styles vs. Character Styles

Word has two different kinds of styles, applying at two different scopes:

- **Paragraph styles** (e.g. `Heading 1`, `Normal`) apply to an **entire paragraph** at once — font, size, colour, spacing before/after, indentation, alignment, all bundled together as one definition.
- **Character styles** (e.g. `Strong`, `Emphasis`) apply only to the **selected run of text** within a paragraph, leaving the rest of that paragraph's own paragraph style completely untouched — bolding a single important word inside an otherwise normal sentence, without affecting the sentence's own alignment or spacing.

The Built-In Heading Styles and Why They're Special

`Heading 1`, `Heading 2`, and `Heading 3` aren't just "bigger, bolder text" presets — Word specifically recognises them as **structural markers** of the document's own outline. A paragraph styled as Heading 1 is understood by Word to be a top-level section; a Heading 2 immediately after it is understood to be a subsection within that Heading 1 — this structural understanding is exactly what powers the **Navigation Pane**, and exactly what Chapter 4's automatic Table of Contents will read from directly.

Applying and Modifying a Style

Clicking anywhere inside a paragraph (no need to select the whole thing) and clicking a style in the **Styles gallery** applies it to that entire paragraph. Right-clicking any style in the gallery and choosing **Modify** opens its definition — font, size, colour, spacing — and confirming a change updates **every paragraph in the document currently using that style**, immediately, without reselecting a single one of them. This is precisely Excel Fundamentals Chapter 6's own Cell Styles behaviour, just applied to paragraphs instead of cells.

The Navigation Pane: Built on Heading Styles

View → Navigation Pane shows a collapsible outline of every Heading-styled paragraph in the document — clicking any entry jumps straight to that section, instantly, in a document that might otherwise be dozens of pages long. This only works because Word can see which paragraphs are genuinely tagged with a Heading style; a paragraph that merely *looks* like a heading (through direct formatting alone) is invisible to the Navigation Pane entirely, no matter how convincingly it's formatted.

	MANUALLY BOLDED & ENLARGED TEXT	THE HEADING 1 STYLE APPLIED
Looks like a heading	Yes	Yes
Appears in the Navigation Pane	No	Yes
Picked up by an automatic Table of Contents (Ch.4)	No	Yes
Updating every heading's look at once	Must reselect and reformat each one individually	Modify the style once — every instance updates

THE EXACT SAME IDEA AS EXCEL'S OWN CELL STYLES, ONE CHAPTER EARLIER IN SPIRIT

Excel Fundamentals Chapter 6 introduced Cell Styles specifically so a design decision (say, changing a colour) could be updated once and apply everywhere that style was used, instead of hunting down every individually-formatted cell. Word's Styles do precisely the same job for paragraphs and text runs — recognising this as the same underlying pattern, reapplied in a new app, matters more here than memorising every individual style name.

A HEADING THAT ONLY "LOOKS" LIKE ONE IS INVISIBLE TO EVERYTHING THAT MATTERS LATER

Manually bolding and enlarging a line of text produces something visually indistinguishable from a real Heading style at first glance — but Word has no way to infer that bold, 18-point text was *intended* as a heading unless the actual Heading style was applied. This isn't a cosmetic detail: it directly breaks the Navigation Pane today, and it will directly break Chapter 4's automatic Table of Contents later — always apply a real Heading style to anything meant to function as a heading, never direct formatting alone.

Hands-On Exercises

EXERCISE 1

In a new Word document, type three section titles, each followed by a short paragraph of body text. Apply the Heading 1 style to all three titles. Open the Navigation Pane and confirm all three appear. Then modify the Heading 1 style's colour. Describe what happens to all three titles at once.

 [View solution](#)

EXERCISE 2

Take one of the three section titles from Exercise 1 and change it back to the Normal style, then manually bold it and increase its font size to visually match the other two headings. Check the Navigation Pane again. Explain exactly what you see and why, even though all three titles still look visually identical on the page.

 [View solution](#)

EXERCISE 3

Within one body paragraph (styled Normal), select a single important word and apply the built-in "Strong" character style to it. Explain what changed about that one word versus what stayed exactly the same about the rest of the paragraph, and why a character style is the correct tool here rather than a paragraph style.

 [View solution](#)

CHAPTER 2 QUICK REFERENCE

- **Direct formatting** — changes only the selected text, no name or reusable meaning attached
- **Style** — a named, reusable formatting definition; modifying it updates every paragraph/run using it, everywhere
- **Paragraph style** (e.g. Heading 1) — applies to the whole paragraph; **character style** (e.g. Strong) — applies only to selected text within it
- **Heading 1/2/3** are structurally special — Word treats them as the document's own outline, not just larger text
- **Navigation Pane** (View menu) — jumps between sections, but only ones tagged with a real Heading style
- Text that merely LOOKS like a heading (bold + large, no Heading style) is invisible to the Navigation Pane and to Chapter 4's automatic Table of Contents

Word & PowerPoint

Chapter 3 · Page Layout: Margins, Page Breaks vs. Section Breaks

Chapter 1 established that Word's content simply flows, with pages existing only as a consequence of that flow rather than as fixed containers. This chapter is about the one genuine exception: page-level formatting — margins, orientation, headers — that Word treats as belonging to a **section**, not a page. Understanding the difference between a page break (which affects nothing but where text starts) and a section break (which can change everything about how a region of the document looks) resolves one of the most common sources of Word frustration.

Margins: The Page's Own Boundaries

Layout → **Margins** sets how much blank space surrounds the text on every page — by default, this single setting applies to the **entire document**, uniformly. There's no way to give one specific page different margins than its neighbours unless the document is divided into separate sections first — margins are a section-level property, not a page-level one, exactly the distinction this chapter builds toward.

Page Breaks: Forcing Content to the Next Page

Ctrl+Enter inserts a manual page break — whatever comes immediately after it is forced to start at the top of the next page, regardless of how much room was left on the current one. This is useful and common (starting a new chapter cleanly, keeping a table from splitting awkwardly across two pages) — but it does **nothing else**. A page break creates no new formatting boundary at all: everything before it and everything after it still belongs to the exact same section, sharing identical margins, orientation, and headers/footers.

The Problem Page Breaks Can't Solve

Suppose one specific page in an otherwise portrait document needs to be landscape — a wide table or chart that genuinely doesn't fit in a portrait page's width. Inserting a page break before that content does nothing to change its orientation; a page break has no concept of "and also change these formatting properties for what follows." Page orientation, like margins, is a property of the *section* a page belongs to — and a page break doesn't create a new section, so every page keeps whatever orientation the one shared section was already set to.

Section Breaks: Genuine Formatting Boundaries

Layout → Breaks → Section Breaks offers something a page break can't: a real division that lets everything after it — margins, orientation, headers/footers, page numbering — differ completely from what came before, while everything *within* a section stays consistent.

SECTION BREAK TYPE	WHAT IT DOES
Next Page	Starts the new section at the top of the next page — the most common choice, functionally like a page break that also opens a new formatting region
Continuous	Starts the new section without forcing a page break at all — useful for changing something mid-page, like switching to a two-column layout for one paragraph and back to one column immediately after
Even Page / Odd Page	Starts the new section on the next even- or odd-numbered page — the traditional print-book convention of always starting a new chapter on a right-hand (odd) page

With a Next Page section break inserted immediately before the wide table from the earlier example, that one section can be set to Landscape (Layout → Orientation, applied while your cursor is inside that specific section) while every other section in the document stays Portrait — solving exactly the problem a plain page break couldn't.

	PAGE BREAK (CTRL+ENTER)	SECTION BREAK
Forces content to a new page	Yes	Yes (Next Page / Even Page / Odd Page types)
Can change margins for what follows	No	Yes
Can change page orientation for what follows	No	Yes
Can give the new region its own header/footer	No	Yes
Can restart or reformat page numbering	No	Yes

THIS CHAPTER'S SECTION BREAKS ARE CHAPTER 5'S OWN PREREQUISITE

Chapter 5 covers giving a long document a different header on its very first page, or different headers for different chapters — none of that is possible without first dividing the document into the sections this chapter introduces. Getting comfortable inserting and recognising section breaks now makes Chapter 5's header/footer material click immediately rather than needing to double back to this one.

A PAGE BREAK WHERE A SECTION BREAK WAS NEEDED LOOKS FINE UNTIL IT DOESN'T

Using Ctrl+Enter (or repeatedly pressing Enter) to "push" content onto its own page is a common substitute for a section break, and it looks completely correct — right up until that content actually needs different margins, orientation, or a header. At that point, changing the formatting "there" silently changes it for the *entire document*, since there's only ever been one section the whole time. Conversely, deleting an existing section break merges the two sections it separated back into one — and the surviving section adopts whichever section's own formatting came *after* the deleted break, which can unexpectedly change how the earlier content looks the moment the break disappears.

Hands-On Exercises

EXERCISE 1

In a new Word document, type a paragraph, insert a page break (Ctrl+Enter), then type another paragraph on the new page. Change the document's margins. Describe what happens to both paragraphs, and explain why the page break made no difference to the result.

 [View solution](#)

EXERCISE 2

You need page 4 of a 6-page portrait report to be landscape, with every other page staying portrait. Describe the exact sequence of section breaks needed (how many, of which type, placed where) and which page(s) you'd then apply the Landscape orientation to.

 [View solution](#)

EXERCISE 3

A document has two sections: Section 1 (portrait) and Section 2 (landscape), divided by a Next Page section break. A colleague deletes that section break to "clean up" the document. Explain exactly what happens to the page(s) that were in Section 1, and why.

 [View solution](#)

CHAPTER 3 QUICK REFERENCE

- **Margins** apply to a whole section, not an individual page — by default, that means the whole document
- **Ctrl+Enter** — a page break; forces a new page, changes nothing else
- **Layout → Breaks → Section Breaks** — Next Page, Continuous, Even Page, Odd Page — each opens a genuinely new formatting region
- Only a section break lets margins, orientation, headers/footers, or page numbering differ across a document
- Deleting a section break merges two sections — the surviving formatting is whichever section came *after* the deleted break

Word & PowerPoint

Chapter 4 · Automatic Tables of Contents, Captions & Cross-References

Chapter 2 made a point of insisting on real Heading styles rather than text that merely looks like a heading, promising the payoff would come later. This chapter is that payoff: an automatic Table of Contents, auto-numbered figure and table captions, and cross-references that all stay correct on their own — provided the structural groundwork from Chapter 2 is actually in place.

Inserting an Automatic Table of Contents

References → **Table of Contents** builds an entry for every paragraph in the document tagged with a Heading style, using each heading's own text and current page number. Crucially, the result is a **field**, not typed text — it looks like ordinary text on the page, but it's actually a live query against the document's own current structure, re-run every time it's updated.

Why Manually Typing a Table of Contents Fails

Typing out "Introduction 3" by hand produces something that looks identical to a real Table of Contents entry — until a paragraph is added earlier in the document and "Introduction" actually starts on page 4 instead. A manually typed entry has no connection to the document's actual content at all; nothing about it updates, and nothing warns you it's now wrong. This is exactly the same trap Chapter 2 warned about with headings that only *look* the part — here the stakes are just higher, since a wrong page number in a real report is a genuinely embarrassing, easily-avoidable mistake.

Updating the Table

Right-click anywhere inside the Table of Contents and choose **Update Field** (or click it and use References → Update Table) to bring it current. Word offers two options:

- **Update page numbers only** — faster; use this when headings themselves haven't changed, only their position on the page has (content was added or removed elsewhere).
- **Update entire table** — slower but complete; required whenever a heading's own text changed, or a heading was added or removed, since the first option won't notice a genuinely new or renamed entry.

Captions: Numbering Figures and Tables Automatically

Select an image or table, then **References** → **Insert Caption**, choosing a label ("Figure," "Table," or a custom one). Word inserts "Figure 1" (or "Table 1," and so on) automatically, numbered by counting how many captions of that same label already exist earlier in the document. Insert a brand-new figure *between* two existing ones, and every caption after it — figures that were "Figure 3," "Figure 4"... — automatically rennumbers to stay sequential, with zero manual editing.

A manually typed "Figure 3" has no such awareness — insert a new figure before it, and it stubbornly stays "Figure 3" even though it's now actually the fourth figure in the document, silently mislabelling everything from that point on.

Cross-References: Pointing at a Specific Item by Name, Not by Guessing

Insert → Cross-reference creates a live reference to a specific heading, caption, or bookmark elsewhere in the document — "see Figure 3" or "see Section 2.1" — that automatically updates its own displayed number if the referenced item's number ever changes, exactly the same live-field behaviour as the caption itself. Typing "see Figure 3" as plain text instead creates the identical trap as a manually-typed caption: accurate the moment it's written, silently wrong the moment anything shifts.

FEATURE	TYPED MANUALLY	INSERTED AS A REAL FIELD
Table of Contents entry	Frozen the instant it's typed; never notices document changes	Reflects the document's real current structure after an Update
Figure/Table caption number	Stays fixed even if new figures are inserted earlier	Renumbers automatically to stay sequential
A "see Figure X" reference	Can silently point at the wrong figure after a renumber	Updates its own number automatically alongside the caption it refers to

NONE OF THIS WORKS WITHOUT CHAPTER 2'S REAL HEADING STYLES

An automatic Table of Contents reads structure the exact same way the Navigation Pane does — from genuine Heading-styled paragraphs, not text that merely resembles a heading. A document full of manually-bolded, enlarged "headings" produces an empty or incomplete Table of Contents no matter how carefully References → Table of Contents is configured, for precisely the reason Chapter 2's own warning box described.

A TABLE OF CONTENTS IS ONLY AS CURRENT AS ITS LAST UPDATE

Every field this chapter covers — the Table of Contents, captions, cross-references — reflects the document's structure as of whenever it was **last updated**, not whenever it's actually viewed or printed. Add a new chapter, insert a figure, or retitle a section, and every one of these fields silently keeps showing its old, now-stale information until explicitly updated — with no on-screen warning that anything has drifted out of sync. Always update every field in a document (Ctrl+A, then F9 updates every field at once) as a genuine last step before printing or exporting to PDF, not an optional afterthought.

Hands-On Exercises

EXERCISE 1

In a document with three Heading-1-styled sections, insert an automatic Table of Contents. Add a brand-new fourth section with its own Heading 1 title somewhere in the middle of the document. Explain exactly what you need to do for the Table of Contents to show all four sections correctly, and which of the two update options is required here specifically.

 [View solution](#)

EXERCISE 2

A document has three images, captioned "Figure 1," "Figure 2," and "Figure 3" using Insert Caption. You insert a new image between the current Figure 1 and Figure 2, and caption it the same way. Describe what happens to the caption numbers of every figure in the document, and contrast this with what would have happened if all three original captions had instead been typed manually as plain text.

 [View solution](#)

EXERCISE 3

A 20-page report has a Table of Contents, several captioned figures, and a few "see Figure X" cross-references, all inserted correctly as real fields. Two days later, three paragraphs are added near the start of the document, and it's exported straight to PDF without reopening the Table of Contents or any fields. Explain exactly what a reader of the PDF would see, and what single step would have prevented it.

 [View solution](#)

CHAPTER 4 QUICK REFERENCE

- **References → Table of Contents** — built from Heading-styled paragraphs only; a live field, not typed text
- **Update Field** — "page numbers only" for content shifts; "entire table" when headings themselves changed
- **References → Insert Caption** — auto-numbered Figure/Table labels that renumber automatically when new ones are inserted
- **Insert → Cross-reference** — a live "see Figure X" reference that updates alongside its target
- Every field reflects the document only as of its **last update** — always update all fields (Ctrl+A, F9) before printing or exporting
- None of this works on headings/captions that only visually resemble the real thing — see Chapter 2

Word & PowerPoint

Chapter 5 · Headers, Footers & Page Numbers in Long Documents

Chapter 3 introduced section breaks without fully cashing in on why they matter — this chapter is where that investment pays off. Headers, footers, and page numbers are all section-level properties, exactly like the margins and orientation Chapter 3 covered, and getting a genuinely professional-looking long document (a cover page with no header, a different chapter title in each section's own header, front matter numbered differently from the main content) depends entirely on the section structure Chapter 3 already taught you to build.

Headers and Footers: The Basics

Insert → **Header** (or **Footer**) adds content that repeats automatically at the top or bottom of every page, opening the **Header & Footer Tools** contextual ribbon tab while you're editing one. Double-clicking directly on a header or footer area on the page jumps straight into editing it the same way.

"Different First Page": A Special Case Needing No Section Break

Header & Footer Tools → **Options** → **Different First Page** lets the very first page of a section have its own header/footer content (often none at all), while every following page in that same section keeps the regular one. This is a simple checkbox, not a new section — useful for a cover page or a chapter's opening page that shouldn't show a running header, without needing the heavier machinery of a section break just for this one common case.

"Different Odd & Even Pages"

Options → **Different Odd & Even Pages** supports the classic double-sided, print-book layout: page numbers positioned on the outside edge of each page (right side on odd/right-hand pages, left side on even/left-hand pages), or a document title on even pages contrasted with a chapter title on odd pages — a detail that only actually matters once a document is genuinely printed and bound double-sided, but a real one for anything intended to be.

Different Headers Per Section: Where Chapter 3 Actually Pays Off

By default, every section's header is set to **"Same as Previous"** — meaning even though Chapter 3's section breaks divide the document into genuinely separate formatting regions, the header still just mirrors whatever the very first section's header contains, everywhere. Clicking into a later section's header and turning off **Link to Previous** (a toggle button in Header & Footer Tools, highlighted when active) is the one action that makes that section's header genuinely independent — only then can it show, say, the current chapter's own title instead of the document's overall title.

Page Numbers: Formats and Restarting the Count

Insert → **Page Number** → **Format Page Numbers** controls both the number style (**1, 2, 3** vs. **i, ii, iii** vs. **A, B, C**) and whether numbering **continues** from the previous section or **restarts** at a chosen value — both settings apply per section, the same way margins and orientation do. The classic use: front matter (a table of contents, a preface) numbered in lowercase roman numerals, with the main content's first page restarting cleanly at Arabic "1" once it begins — a section break between the two is what makes both the numbering style and the restart point possible at all.

SETTING	WHAT IT ACTUALLY SOLVES
Different First Page	The first page of a section needs no header/footer (or a different one) — no new section required
Different Odd & Even Pages	Double-sided printing — mirrored header/footer content and page-number position per side
Link to Previous (turned OFF)	A specific section needs a genuinely independent header/footer, distinct from every other section
Format Page Numbers — restart	Front matter and main content need separate numbering styles and starting points

LINK TO PREVIOUS IS THE SINGLE SETTING THAT MAKES CHAPTER 3'S SECTIONS WORTH HAVING

Every section break Chapter 3 taught you to insert accomplishes nothing for headers specifically until Link to Previous is switched off for the section that needs its own content. It's easy to correctly insert three section breaks, confirm each section has its own margins or orientation, and then wonder why every header in the document still says the same thing — the answer is almost always that Link to Previous was never turned off anywhere.

EDITING A "LINKED" HEADER EDITS EVERY OTHER LINKED SECTION'S HEADER TOO, SILENTLY

If Link to Previous is still ON for a section, clicking into that section's header and typing a change doesn't just affect that one section — it changes the header for every section still linked back to the same original source, since they're all genuinely displaying the identical shared content, not independent copies of it. This is easy to miss because nothing in the interface loudly announces "you're editing a shared header right now" — the toggle button's highlighted/not-highlighted state is the only visual cue, and it's easy to overlook while focused on the header text itself.

Hands-On Exercises

EXERCISE 1

A document has a cover page followed by the main report content, all in one section. Describe exactly which single setting removes the header from just the cover page while keeping it on every following page, and explain why this doesn't require inserting a section break.

 [View solution](#)

EXERCISE 2

A report has three Chapter sections (via Next Page section breaks from Chapter 3), and each chapter's header should show that specific chapter's own title. Currently every header shows "Chapter 1" regardless of which section you're viewing. Describe exactly what's missing and the steps to fix Chapter 2 and Chapter 3's headers specifically.

 [View solution](#)

EXERCISE 3

A document has front matter (roman numerals i, ii, iii) followed by main content that should restart at Arabic "1". Describe the section structure needed and exactly which Format Page Numbers settings apply to which section.

 [View solution](#)

CHAPTER 5 QUICK REFERENCE

- **Different First Page** — a checkbox, not a section break; removes/changes just the first page's own header/footer
- **Different Odd & Even Pages** — mirrored content for double-sided printing
- **Link to Previous** (Header & Footer Tools) — must be turned OFF for a section to get a genuinely independent header/footer; ON by default, mirroring the previous section
- **Format Page Numbers** — style (1/i/A) and restart point are both per-section settings
- Editing a header while Link to Previous is still ON silently edits every other linked section too
- None of this is possible without Chapter 3's section breaks already dividing the document

Word & PowerPoint

Chapter 6 · Track Changes and Comments: Real Collaboration in Word

Every chapter so far assumed a single author shaping one document alone. Real documents get reviewed, edited, and argued over by more than one person — and Chapter 1's own flowing-text model, where content has no fixed address to point at, makes that genuinely risky without the right tooling. Track Changes and Comments are Word's answer: a record of exactly who changed what, and a way to discuss content without silently altering it.

Turning On Track Changes

Review → **Track Changes** (or **Ctrl+Shift+E**) toggles recorded editing on. Once active, nothing you type or delete is applied directly to the document's actual content — every insertion and deletion is instead recorded as its own distinct, visible, reversible edit.

Reading Tracked Changes

Inserted text appears underlined; deleted text appears struck-through rather than actually disappearing — both in a colour Word assigns automatically per reviewer, so several people's edits stay visually distinguishable from each other in the same document. Hovering over any tracked change shows exactly who made it and when, turning a shared document into a genuine, attributable edit history rather than an anonymous pile of changes.

Accepting and Rejecting Changes

Review → **Accept** or **Reject**, either one change at a time or via Accept All/Reject All. Accepting a tracked insertion makes it permanent, ordinary text; accepting a tracked deletion actually removes the struck-through text for good. Rejecting either one reverses it completely, restoring the document to exactly how it read before that specific change was proposed.

Comments: Discussion Without Changing the Text

Review → **New Comment** attaches a note to a specific selection, shown in the margin, without altering the document's actual content in any way. Comments can be replied to (threading a small discussion under the original note) and marked **Resolve** once addressed — which greys the comment out without deleting it, preserving the discussion for later reference rather than erasing it outright.

Comments vs. Tracked Changes: Genuinely Different Purposes

A **Comment** is a question or note ABOUT the content — "should this figure be updated?" — that never touches the actual text. A **Tracked Change** is an actual proposed edit TO the text itself — replacing a word, adding a sentence — that can subsequently be accepted or rejected. Using a comment to say "change this to X" instead of just making the tracked change directly adds an unnecessary extra round trip; using a tracked change for a pure question ("why is this here?") leaves an edit sitting in the document with nothing genuine to accept or reject.

Review Views: Simple Markup, All Markup, No Markup, Original

The Review tab's own view dropdown controls how tracked changes are displayed, without affecting whether they've actually been accepted:

- **All Markup** — every insertion and deletion shown inline, exactly as described above.
- **Simple Markup** — a cleaner read, with a small vertical line in the margin marking where changes exist, without cluttering the text itself.
- **No Markup** — a preview of how the document would look if every currently-tracked change were *accepted* — without actually accepting a single one.
- **Original** — the reverse preview: how the document would look if every currently-tracked change were *rejected*.

Comparing Two Versions of a Document

Review → Compare takes two separate `.docx` files — say, the version you originally sent out, and a colleague's edited copy that was worked on with Track Changes switched off the entire time — and generates a brand-new document showing every difference between them as proper tracked changes, computed automatically. This solves exactly the situation Track Changes alone can't: recovering a real edit history from two files where no tracking ever actually happened.

TOOL	SOLVES
Track Changes	Recording actual proposed edits to the text, live, as they're made
Comments	Discussing content without touching the text at all
Compare	Reconstructing an edit history after the fact, from two untracked files

TRACK CHANGES IS WORD'S ANSWER TO SAFELY SHARING A FLOWING DOCUMENT

Chapter 1 established that Word has no independently addressable units the way Excel's cells do — everything is one continuous stream. That's exactly why simultaneous, untracked edits from more than one person are so hard to reconstruct afterward: there's no cell-level history to fall back on. Track Changes exists specifically to make that flowing model safe to share, by keeping every edit individually attributable and reversible instead of silently merged into the stream.

"NO MARKUP" VIEW CAN LOOK DECEPTIVELY FINISHED

Switching to No Markup view to preview a clean-looking result is easy to mistake for actually having accepted every change — it hasn't. The changes are still fully tracked and pending underneath; if the file is saved, sent, and reopened by someone else viewing it in All Markup, every single tracked edit is still exactly where it was, unaccepted. Separately: forgetting to turn Track Changes on *before* starting a collaborative editing session applies every change directly and irreversibly — there is no way to recover which parts were original versus edited after the fact, short of Review → Compare against a separately-saved earlier copy, if one happens to exist.

Hands-On Exercises

EXERCISE 1

Turn on Track Changes, then delete a sentence and add a new one in its place. Switch between All Markup, Simple Markup, No Markup, and Original views without accepting or rejecting anything. Describe what each of the four views shows, and confirm the underlying tracked changes are still fully intact in every one of them.

 [View solution](#)

EXERCISE 2

A colleague asks, via a Comment, "should this paragraph mention the Q3 numbers?" rather than just editing the paragraph directly. Explain why a Comment was the appropriate tool here rather than a Tracked Change, and describe what would have been lost if they'd instead just deleted the paragraph as a tracked change with no explanation.

 [View solution](#)

EXERCISE 3

You emailed a colleague a report. They made several edits with Track Changes switched off, then emailed the edited file back under a new filename. Describe exactly how to see what they actually changed, and explain why simply opening their file and turning Track Changes on now would not reveal their edits.

 [View solution](#)

CHAPTER 6 QUICK REFERENCE

- **Ctrl+Shift+E** — toggle Track Changes; records insertions/deletions instead of applying them directly
- **Accept/Reject** — makes a tracked change permanent, or fully reverses it
- **Comments** — discussion attached to a selection, never alters the actual text
- **All Markup / Simple Markup / No Markup / Original** — four ways to VIEW tracked changes; none of them accept or reject anything
- **Review** → **Compare** — generates tracked changes between two separately-saved files, after the fact
- Untracked collaborative edits are irreversible once made — there's no cell-level fallback the way Excel has

Word & PowerPoint

Chapter 7 · Mail Merge: Generating Many Personalized Documents from One Template

Every prior chapter treated a Word document as one single, self-contained thing. Mail Merge turns that idea inside out: one document becomes a **template**, and an external table of data — most naturally, an Excel worksheet — supplies as many personalized versions of it as there are rows. This is also the most direct connection point between the two apps this course covers: the data source Mail Merge reads is exactly the kind of Excel Table Excel Fundamentals Chapter 7 taught you to build.

The Problem Mail Merge Solves

Imagine sending a personalized letter to 200 people — each one needs their own name, address, and a specific amount owed. Manually creating 200 separate documents, or find-and-replacing placeholder text 200 times in one document, is exactly the kind of repetitive, error-prone task that shouldn't be done by hand. Mail Merge produces all 200 personalized documents from one template and one data table, correctly and automatically.

The Two Ingredients: A Template and a Data Source

A merge always needs exactly two things:

- A **template** — a normal Word document, written once, with placeholder fields marking where personalized content goes.
- A **data source** — a table of records, one row per recipient, one column per piece of information. Most commonly, this is an Excel worksheet — and ideally, an actual Excel Table (Excel Fundamentals Chapter 7), whose column headers become the exact field names Mail Merge offers.

Setting Up the Merge

Mailings → **Start Mail Merge**, choosing the document type (Letters, Envelopes, Labels, or Email Messages), then **Select Recipients** → **Use an Existing List**, browsing to the Excel workbook and picking the specific worksheet or named Table it should read from.

Inserting Merge Fields

Mailings → **Insert Merge Field** lists every column found in the connected data source — inserting one drops a placeholder like «FirstName» or «Amount» directly into the template at the cursor's position. **Preview Results** toggles between seeing these raw placeholders and an actual preview populated with one real row's data, letting you confirm the layout looks right before running the full merge.

Matching Fields: The Same Exact-Match Discipline as XLOOKUP

Mailings → **Match Fields** exists because Word's own expected field names (like "First Name") don't always exactly match whatever the Excel source's actual column headers are called (say, "FName" instead). This dialog lets you map Word's expected names onto the source's real column names explicitly — skip this when they genuinely don't match, and merge fields come through blank, or paired with entirely the wrong column, with no error shown.

This is precisely the same discipline Excel Fundamentals Chapter 5 demanded of XLOOKUP: an exact match is required for a lookup to succeed, and a near-miss (different capitalisation, an extra space, a slightly different name) fails silently rather than obviously — just applied here across a whole template's worth of field names at once, instead of one single lookup key.

Completing the Merge

Mailings → **Finish & Merge** offers three outputs: **Edit Individual Documents** (combines every recipient's personalized version into one new document, one after another), **Print Documents** (sends the entire batch straight to a printer), or **Send Email Messages** (for an Email Messages-type merge, sending each personalized version directly to that row's own email address).

MAIL MERGE CONCEPT	ITS EXCEL FUNDAMENTALS COUNTERPART
The data source (rows/columns)	An Excel Table (Chapter 7) — one row per record, named columns
A merge field's column name	A Table's own column header, used the same way a structured reference names a column
Match Fields' exact-name requirement	XLOOKUP's own exact-match discipline (Chapter 5)
A blank cell silently merging as empty	AVERAGE silently excluding a blank rather than treating it as zero (Chapter 4)

AN EXCEL TABLE IS THE DIRECT BRIDGE BETWEEN THE TWO HALVES OF THIS COURSE

Everything Excel Fundamentals Chapter 7 taught about a well-structured Table — clear column headers, no blank rows scattered through the data, one record per row — matters just as much here as it did feeding a PivotTable. The better-structured the Excel Table, the more reliably Mail Merge's field matching and personalized output actually work; a messy, inconsistent source table causes exactly the same kind of silent problems in a merge that it would in an Excel PivotTable.

A BLANK CELL IN THE DATA SOURCE DOESN'T CAUSE AN ERROR — IT MERGES AS BLANK, SILENTLY

A recipient row with a missing value in some column doesn't stop the merge or flag anything — that recipient's document simply shows nothing where that field should be, with no warning that anything is wrong. This is the same "no error, just a quietly wrong result" pattern this course has flagged repeatedly on the Excel side (an unlocked reference, an approximate-match lookup) — always spot-check a sample of merged documents, not just the template and the first preview row, before printing or sending 200 personalized copies of a document with a silently blank field.

Hands-On Exercises

EXERCISE 1

You have an Excel Table with columns Name, Amount, DueDate, one row per customer. Set up a Letters-type mail merge connected to this table, insert merge fields for all three columns into a template letter, and preview the result using the first row's data. Describe exactly what you'd see in the preview versus the raw template.

 [View solution](#)

EXERCISE 2

Your Excel data source's columns are named CustName, AmtDue, and Due_Date — not exactly matching the field names your template expects (Name, Amount, DueDate). Explain what would happen if you inserted merge fields and ran the merge without using Match Fields first, and describe the correct fix.

 [View solution](#)

EXERCISE 3

A 300-row Excel data source has one row (row 157) with a blank Amount cell, due to a data-entry mistake. You run the full merge and print all 300 letters without spot-checking. Describe exactly what recipient 157's letter looks like, and what habit from this chapter would have caught the problem before printing.

 [View solution](#)

CHAPTER 7 QUICK REFERENCE

- **Template** (the Word document with placeholders) + **data source** (an Excel Table — one row per recipient) = a merge
- **Mailings** → **Start Mail Merge** → **Select Recipients** connects the data source
- **Insert Merge Field** — inserts a «Column Name» placeholder; **Preview Results** shows it filled with real data
- **Match Fields** — maps Word's expected field names onto the source's actual column headers; skipping this when names differ causes silent blanks or mismatches, the same exact-match discipline XLOOKUP required
- **Finish & Merge** — Edit Individual Documents, Print Documents, or Send Email Messages
- A blank source cell merges as blank silently — always spot-check several merged documents, not just the first preview

Word & PowerPoint

Chapter 8 · Slide Layouts and the Slide Master: Consistency Without Repetitive Work

Chapter 1 promised that Chapter 2's Word Styles and this chapter's Slide Master both solve the exact same problem — define formatting once, apply it everywhere, update it centrally later — just expressed through each app's own content model. Word does it at the level of a paragraph; PowerPoint does it at the level of an entire slide's layout. This chapter delivers on that promise.

Direct Formatting vs. the Slide Master: The Same Idea, Applied to Whole Slides

Manually recolouring a title on one slide changes exactly that slide and nothing else — the PowerPoint equivalent of Chapter 2's direct text formatting. The **Slide Master** instead defines a shared design — fonts, colours, background, where a title and content actually sit — that every slide built from it inherits automatically. Change the Master once, and every slide using it updates together, exactly like editing a Word Style updates every paragraph tagged with it.

Slide Layouts vs. the Slide Master

PowerPoint actually has two levels here, not one. The **Slide Master** itself is the top-level design — the overall font family, accent colour, and background every slide ultimately inherits from. Beneath it sit individual **Slide Layouts** ("Title Slide," "Title and Content," "Two Content," and others) — variations that inherit the Master's design but define their own specific arrangement of placeholders, suited to a particular kind of slide.

Entering Slide Master View

View → **Slide Master** opens a thumbnail pane showing the Master slide at the top, with every Layout indented beneath it. Editing the top Master slide changes something shared by every Layout below it; editing one specific Layout only affects slides that are actually using that particular Layout.

What Belongs on the Master vs. a Specific Layout

Anything every single slide should share — the overall font, the accent colour, a logo or footer text that should appear on every slide regardless of type — belongs on the **Master** itself. Anything that only makes sense for one specific kind of slide — a Title Slide's large centred title with no body content, versus a Title and Content layout's smaller title plus a full content area beneath it — belongs on that specific **Layout** instead, not duplicated across every layout individually.

Placeholders: The Master's Real Building Blocks

A **placeholder** on the Master or a Layout defines where a title, body content, or footer will sit, and its default formatting — but it holds no actual content itself. An individual slide fills a placeholder with its own real text or image, while the placeholder's own position, size, and default formatting continue coming from whichever Layout that slide is using. This is PowerPoint's version of the same separation Chapter 2 introduced: the formatting definition lives centrally (on the Layout/Master), the actual content lives on the individual instance (the slide).

Applying a Different Layout to an Existing Slide

Home → **Layout** lets you switch an existing slide to a different Layout — its actual text content is preserved, but reflows into the new Layout's own placeholder positions and formatting. This is meaningfully different from manually dragging text boxes around on one slide by hand: switching Layouts re-applies a genuine, centrally-defined arrangement, while manually repositioning content on a single slide is a one-off change that affects nothing else and isn't reusable anywhere.

	WORD STYLES (CH.2)	POWERPOINT SLIDE MASTER
Applies to	A paragraph or a run of text	An entire slide's layout and placeholders
Direct formatting equivalent	Manually bolding/resizing text	Manually moving/recolouring one slide's own title box
Editing the shared definition	Modify the style once	Edit the Master or a specific Layout once
What updates automatically	Every paragraph tagged with that style	Every slide built from that Layout (or every Layout, if the Master itself changes)

SAME UNDERLYING GOAL, A DIFFERENT UNIT OF FORMATTING

Word's Styles operate at the level of a paragraph because that's the natural unit inside a flowing document (Chapter 1). PowerPoint's Slide Master operates at the level of a whole slide's placeholder layout because a slide — not a paragraph — is PowerPoint's own natural, independent unit. Recognising "define once, apply everywhere, update centrally" as the shared goal underneath two very differently-shaped features is worth more than memorising either one as a standalone trick.

**MANUALLY OVERRIDING A PLACEHOLDER BREAKS ITS CONNECTION TO FUTURE MASTER EDITS
— EXACTLY LIKE CHAPTER 2'S DIRECT FORMATTING**

Manually recolouring one slide's title text box directly (rather than through the Master or Layout) creates the exact same trap Chapter 2 warned about: that one slide's title is now disconnected from the shared definition for that specific property. Later, changing the Master's title colour updates every other slide correctly — except that one, which keeps whatever colour was manually applied, since direct formatting always takes precedence over whatever the Master or Layout would otherwise supply.

Hands-On Exercises

EXERCISE 1

Open Slide Master view. Change the Master's title font colour. Confirm which slides in your actual presentation update as a result, and explain why editing the Master (rather than one specific Layout beneath it) affects all of them.

 [View solution](#)

EXERCISE 2

On one specific slide, manually change the title text's colour directly (not through Slide Master view). Then go back into Slide Master view and change the Master's title colour again to something different. Describe what happens to that one slide's title versus every other slide's title, and explain why.

 [View solution](#)

EXERCISE 3

A slide currently uses the "Title Slide" layout, with a title typed into it. Apply the "Title and Content" layout to that same slide instead (Home > Layout). Describe what happens to the title text you already typed, and explain why this is meaningfully different from manually dragging the title text box to a new position on the same slide.

 [View solution](#)

CHAPTER 8 QUICK REFERENCE

- **Slide Master** — the top-level shared design (fonts, colours, background) every Layout inherits from
- **Slide Layouts** — variations (Title Slide, Title and Content, etc.) that inherit the Master but define their own placeholder arrangement
- **View** → **Slide Master** — edit the Master (affects everything) or one specific Layout (affects only slides using it)
- **Placeholders** hold no content themselves — their position/formatting comes from the Layout/Master; individual slides only fill them in
- **Home** → **Layout** — switches a slide's layout while preserving its content, reflowing it into the new arrangement
- Manually overriding a placeholder's formatting on one slide disconnects it from future Master/Layout edits — the same trap as Chapter 2's direct formatting

Word & PowerPoint

Chapter 9 · Effective Slide Design: Content Density, Visual Hierarchy, and "Death by Bullet Points"

Chapter 8 covered how to build slides consistently. This chapter is about a genuinely different question: what actually belongs on one. A slide isn't read start to finish the way a Word document is (Chapter 1) — it's glanced at for a few seconds while a presenter talks over it. Content that would be perfectly reasonable in a report becomes a liability on a slide, and this chapter is about recognising the difference.

The "Death by Bullet Points" Problem

A slide crammed with six or more bullet points, each a full sentence, effectively becomes a script the presenter reads aloud — at which point the audience faces a genuine dilemma: read the slide, or listen to the speaker, since doing both well at the same time is difficult. This split attention measurably reduces how much of *either* the audience actually retains, defeating the purpose of using a visual medium at all.

The 6×6 Guideline

A widely-used rule of thumb: no more than roughly **6 bullet points** per slide, no more than roughly **6 words** per bullet. This isn't a strict law — plenty of good slides break it in one direction or another — but it's a genuinely useful gut-check against the instinct to fit "just one more point" onto an already-busy slide.

One Idea Per Slide

A slide should communicate **one main point**. If a slide is trying to cover several genuinely distinct ideas, the right fix is almost always splitting it into several slides rather than compressing everything onto one. This costs nothing structurally in PowerPoint the way it might in a Word document — Chapter 1 established that a slide is a discrete, independent unit, so adding a second or third slide to cover what one overcrowded slide was trying to do doesn't disrupt anything the way inserting extra pages into flowing Word content might.

Visual Hierarchy: Guiding the Eye on Purpose

Font size, weight, colour, and position should reflect actual importance. A slide's title should be unmistakably the single most prominent thing on it; supporting details should visibly read as secondary, through smaller size or lighter weight, not compete with the title for attention. A slide where every line of text is the same size and weight forces the audience to guess what actually matters, rather than being shown.

Text vs. Visuals: When a Chart or Image Says It Better

Excel Fundamentals Chapter 8 established that a chart type should be chosen based on the actual question being asked of the data, not habit. The same principle applies to slide content generally: "sales grew 40% across four regions" communicates far faster as a single simple bar chart than as a paragraph of prose or a wall of bullet points trying to describe the same numbers in words. Whenever a slide is struggling to explain something numeric or comparative in text, the real fix is usually a chart, not a longer bullet list.

Whitespace: Empty Space Is Doing Work, Not Wasted

Filling every inch of a slide can feel like using the space efficiently, but empty space around content is what actually lets the eye rest and land on what matters. A slide with generous margins around one clear focal point reads faster and more confidently than a dense slide packed edge to edge, even when the dense slide technically contains useful information — the audience simply can't tell what to look at first.

DIMENSION	CROWDED SLIDE	EFFECTIVE SLIDE
Bullet count	8-10+, full sentences	Roughly 6 or fewer, short phrases
Ideas covered	Several distinct points crammed together	One clear idea, split across more slides if needed
Numeric/comparative data	Described in a paragraph or bullet list	Shown as a chart matched to the actual question (Excel Ch.8)
Visual hierarchy	Every line the same size/weight	Title unmistakably dominant; details visibly secondary
Use of empty space	Filled edge to edge	Generous margins around one clear focal point

CHAPTER 8 IS THE "HOW," THIS CHAPTER IS THE "WHAT"

Slide Master and Slide Layouts (Chapter 8) make sure whatever content you put on a slide looks visually consistent with every other slide. This chapter is about deciding what that content should actually be in the first place — the two are genuinely complementary rather than overlapping, and a beautifully consistent deck can still fail if every individual slide is still crammed with too much text.

A SLIDE IS NOT A SUBSTITUTE FOR A LEAVE-BEHIND DOCUMENT

A common instinct is to cram every relevant detail onto slides "just in case" someone reads the deck later without the presenter there to explain it — but the fix for that real need is a genuine, separate Word document (exactly what Chapters 2-7 of this course cover), or PowerPoint's own Notes Page for the presenter's private reference (Chapter 11 covers this), not more visible bullet points crammed onto the live presentation itself. Confusing "what the audience sees during the talk" with "what a reader needs afterward without you there" is one of the most common causes of an overcrowded deck.

Hands-On Exercises

EXERCISE 1

A slide has 9 bullet points, each a full sentence, covering three genuinely separate topics (background, current results, and next steps). Describe how you'd restructure this using the "one idea per slide" principle, and explain why splitting it doesn't cost anything structurally the way inserting new pages into a Word document mid-flow might.

 [View solution](#)

EXERCISE 2

A slide currently reads: "Revenue increased in Q1 by 12%, in Q2 by 18%, in Q3 by 9%, and in Q4 by 24%, showing an accelerating upward trend across the year." Explain why this is a poor fit for a slide as written, and describe specifically what you'd replace it with instead, referencing the relevant principle from Excel Fundamentals Chapter 8.

 [View solution](#)

EXERCISE 3

A colleague insists on cramming extensive detail onto every slide because "people need to be able to read this later if they missed the meeting." Explain what's actually being conflated here, and describe the two correct alternatives this chapter offers for meeting that real need without overcrowding the live presentation.

 [View solution](#)

CHAPTER 9 QUICK REFERENCE

- **6×6 guideline** — roughly 6 bullets per slide, 6 words per bullet; a gut-check, not a hard rule
- **One idea per slide** — split overcrowded slides rather than compressing several ideas onto one
- **Visual hierarchy** — size/weight/position should reflect actual importance, title unmistakably dominant
- Numeric or comparative content usually communicates faster as a **chart** (Excel Ch.8's own framework) than as text
- **Whitespace** around a focal point helps the eye find what matters — a slide doesn't need every inch filled
- A leave-behind need is solved with a real Word document or the Notes Page (Ch.11) — not more visible bullets on the live slide

Word & PowerPoint

Chapter 10 · Animations and Transitions, Used Sparingly

Chapter 9 argued for restraint with text and bullet density. Motion deserves exactly the same scrutiny — animations and transitions are genuinely useful tools that are just as easy to overuse as bullet points, and for the same underlying reason: anything competing for the audience's attention makes the actual content harder to follow, not easier.

Animations vs. Transitions: Two Different Things

An **animation** applies motion to one element *within* a single slide — a bullet point flying in, a chart bar growing into place. A **transition** is the effect used when moving *from* one slide *to* the next — a fade, a wipe, a push. Different scope entirely: animations live inside a slide, transitions live between slides.

The Three Animation Categories

The Animations tab groups every effect into three colour-coded categories:

- **Entrance** (green) — how an element first appears: Fade In, Fly In, Appear.
- **Emphasis** (yellow) — draws attention to something already visible: Pulse, a colour change, a brief grow/shrink.
- **Exit** (red) — how an element leaves: Fade Out, Fly Out, Disappear.

Applying and Sequencing Animations

Select an element, then pick an effect from the Animations tab. Multiple animations on one slide play in a numbered order, shown as small numbered tags beside each animated element. The **Animation Pane** (Animations → Animation Pane) lists every animation on the current slide in that same order, letting you drag to reorder them or adjust each one's own timing directly.

Start Triggers: On Click, With Previous, After Previous

Every animation has a start trigger controlling when it plays:

- **On Click** — waits for the presenter to click before playing, putting pacing directly under the presenter's control.
- **With Previous** — plays automatically, at the same moment as whatever animation came before it.
- **After Previous** — plays automatically, immediately once the previous animation finishes, with no click needed.

Transitions Between Slides

The **Transitions** tab applies an effect to the change between the current slide and the next, with its own Effect Options (direction, variant) and Duration (speed). **Advance Slide** offers the same underlying choice an animation's start trigger does: "On Mouse Click" (presenter-controlled) versus "After" a set delay (fully automatic, useful for a self-running kiosk-style presentation with nobody clicking through it).

Using Animation to Reveal a Bullet List Progressively

One genuinely useful case: setting each bullet point's own Entrance animation to **On Click**, so bullets appear one at a time as the presenter actually discusses each one, rather than the whole list appearing at once and the audience reading ahead of whatever's currently being said. This directly reinforces Chapter 9's "don't let the slide compete with the speaker" principle — used this way, animation is solving a genuine attention problem, not decorating the slide.

USE OF MOTION	PURPOSEFUL	DECORATIVE OVERUSE
Bullet list reveal	One bullet appears per click, matching the presenter's own pace	Every bullet flies in from a different direction with a different effect
Chart or diagram	Builds up piece by piece to match the explanation	Spins, bounces, or pulses for no reason connected to the content
Slide transitions	One simple, consistent style applied to every slide	A different flashy transition chosen per slide

THE SAME JUDGMENT CALL AS CHAPTER 9, JUST APPLIED TO MOTION INSTEAD OF TEXT

Ask the same question Chapter 9 asked about bullet points: does this animation genuinely serve the audience, or is it just filling space and looking busy? The safest default is picking one simple, consistent entrance effect (or none at all) and one consistent transition style, applied via "Apply to All" — reserving anything more elaborate for the rare case where motion itself is actually part of the explanation (a progressive reveal, a chart genuinely building up).

EXCESSIVE ANIMATION IS A WELL-DOCUMENTED DISTRACTION, NOT A NEUTRAL STYLISTIC CHOICE

Too much motion competes directly with the content itself for the audience's attention — the exact same underlying problem Chapter 9 described for text-dense slides, just expressed as movement instead of words. Separately, elaborate custom animation timings can behave unpredictably when a presentation is opened in a different program or a different version of PowerPoint than the one it was built in — a real practical risk worth remembering for anything that might be shared or presented from someone else's machine.

Hands-On Exercises

EXERCISE 1

A slide has 4 bullet points. Set each one's Entrance animation to Fade In with an On Click trigger. Describe exactly what happens when you advance through the slide during a live presentation, and explain why On Click (rather than After Previous) is the right trigger choice for this specific use case.

 [View solution](#)

EXERCISE 2

You want an unattended kiosk display to cycle through 6 slides automatically, with nobody clicking through them, spending about 8 seconds on each slide. Describe exactly which Transitions setting accomplishes this, and explain why "On Mouse Click" would be the wrong choice here.

 [View solution](#)

EXERCISE 3

A colleague's deck gives every single bullet point across 20 slides its own unique entrance effect (some flying in from the left, others spinning, others bouncing), plus a different transition style between every pair of slides. Explain what's specifically wrong with this approach, referencing the underlying principle this chapter shares with Chapter 9.

 [View solution](#)

CHAPTER 10 QUICK REFERENCE

- **Animation** — motion within a slide; **Transition** — the effect between slides
- **Entrance / Emphasis / Exit** — the three animation categories, colour-coded green/yellow/red
- **Animation Pane** — the full ordered list of a slide's animations, reorderable
- **On Click** (presenter-paced) vs. **With Previous / After Previous** (automatic)
- **Advance Slide: On Mouse Click** vs. **After** a timed delay — the transition-level equivalent of a start trigger
- Use animation to solve a real attention problem (a progressive bullet reveal) — not to decorate every element differently

Word & PowerPoint

Chapter 11 · Presenting: Presenter View, Rehearse Timings, and Exporting

Chapters 8 through 10 covered building a deck that's consistent, well-designed, and appropriately restrained with motion. This chapter is about what happens once it's actually time to present it — and delivers directly on Chapter 9's own promise that a proper place exists for extra detail that shouldn't clutter the visible slide.

Presenter View: What the Presenter Sees vs. What the Audience Sees

Slide Show → **Use Presenter View**, on a dual-monitor setup (a laptop plus a projector or external display), shows the audience only the current slide, full-screen, exactly as designed. Meanwhile the presenter's own screen shows considerably more: the current slide, a preview of the *next* slide, speaker notes, and an elapsed-time clock — all genuinely private, visible only to the person presenting.

Speaker Notes: Where Chapter 9's "Extra Detail" Actually Belongs

View → **Notes** (or the Notes pane directly beneath a slide in Normal view) attaches private text to each individual slide — full sentences, extra data, reminders of what to say — visible only to the presenter in Presenter View, or when printed separately as reference handouts. This is precisely the answer Chapter 9's own closing warning box promised: detailed content that shouldn't be visible during the live presentation has a genuine home here, rather than being crammed onto the slide itself.

Rehearse Timings

Slide Show → **Rehearse Timings** runs through the entire deck in full presentation mode while recording exactly how long you actually spend on each slide. This serves two purposes: practicing pacing before the real thing, and — genuinely useful — automatically setting each slide's own timed advance duration based on a real rehearsal, rather than guessing a number by hand the way Chapter 10's own timed-transition setting otherwise requires.

Setting Up the Slide Show

Slide Show → **Set Up Slide Show** offers three modes: **Presented by a speaker** (full screen, the standard live-presentation mode this chapter has assumed throughout), **Browsed by an individual** (opens in a window rather than full screen, letting a viewer navigate it themselves like a document), and **Browsed at a kiosk** (full screen, loops automatically with no manual input) — this last one is exactly the unattended-display scenario Chapter 10's own exercise assumed, and this is where it's genuinely configured.

Exporting: Video and PDF Handouts

File → Export → Create a Video bakes every timing, transition, and animation into one self-contained video file — shareable with anyone who couldn't attend live and doesn't have PowerPoint installed at all, since the result is just a standard video. **File → Export → Create Handouts** (or **Print → Handouts**) instead produces a static, printable reference — multiple slides per page, optionally with space alongside for notes — PowerPoint's own built-in version of the "leave-behind reference" Chapter 9 also suggested solving with a separate Word document.

FORMAT	GOOD FOR	WHAT'S LOST
The live .pptx deck	Presenting in person, with full animation/transition control	Requires PowerPoint (or a compatible viewer) and the presenter actually being there
Exported video	Sharing with people who missed the live session, no software needed	No interactivity — plays back exactly as recorded, cannot be navigated freely
PDF/printed handouts	A quick static reference, pre-reading, or a leave-behind	No animation, no transitions, no narration — just the final static content

THIS CHAPTER IS CHAPTER 9'S OWN PROMISE, PAID OFF IN FULL

Chapter 9 argued that extensive detail belongs somewhere other than the visible slide — either a real Word document, or PowerPoint's own Notes Page. This chapter is where that second option becomes concrete: Speaker Notes plus Presenter View is precisely the mechanism that keeps a slide clean for the audience while still giving the presenter everything they actually need close at hand.

TEST PRESENTER VIEW ON THE ACTUAL EQUIPMENT BEFORE PRESENTING LIVE

A dual-monitor setup that isn't configured correctly (screens mirrored instead of extended, or the wrong display selected as "primary") can cause the audience's screen to show the presenter's own notes, next-slide preview, and timer instead of the clean full-screen slide — a genuinely common and visible mistake, best caught during a quick test on the actual projector or monitor beforehand, not discovered live in front of the audience. Separately: an exported video's file size scales with resolution and length, and can become large enough to be impractical to email directly to a wide distribution list — worth checking before sending.

Hands-On Exercises

EXERCISE 1

A slide's visible content is a single chart with no on-slide explanation. Add detailed Speaker Notes explaining what the chart shows and what to say about it. Explain exactly who can see this text and under what circumstances, and why writing this detail as Speaker Notes was the right call rather than adding it as visible bullet points on the slide itself.

 [View solution](#)

EXERCISE 2

You need a 10-slide presentation to run unattended at a trade show booth, looping continuously with no one clicking through it. Describe the two things you'd set up (one from this chapter, one from Chapter 10) to achieve this, and explain what would go wrong if you left the slide show mode at its default "Presented by a speaker" setting.

 [View solution](#)

EXERCISE 3

Several colleagues couldn't attend a live presentation and have no copy of PowerPoint installed on their laptops. Another colleague wants a quick printed reference to skim before a related meeting next week. Recommend the appropriate export option for each person, and explain what each option preserves and what it loses compared to the live presentation.

 [View solution](#)

CHAPTER 11 QUICK REFERENCE

- **Presenter View** — current slide, next-slide preview, Speaker Notes, and a timer for the presenter; audience sees only the clean current slide
- **Speaker Notes** (View → Notes) — private per-slide detail, visible only to the presenter or in printed handouts
- **Rehearse Timings** — records real per-slide durations from a practice run; can set Chapter 10's own timed advance automatically
- **Set Up Slide Show** — Presented by a speaker / Browsed by an individual / Browsed at a kiosk (unattended, looping)
- **Export → Create a Video** — one self-contained file, no PowerPoint needed to view it, no interactivity
- **Export → Create Handouts** — a static, printable leave-behind reference
- Always test Presenter View on the actual presentation equipment beforehand — a monitor misconfiguration can expose private notes to the audience

Word & PowerPoint

Chapter 12 · Capstone: A Business Report in Word and Its Companion Deck in PowerPoint

Chapter 1 opened this course with two genuinely different content models under one shared interface. Eleven chapters later, this capstone puts both halves to work on the same real project: a properly structured, personalized Word report, and a disciplined, well-designed PowerPoint deck summarizing it — built, reviewed, and delivered using every tool this course covered.

The Project Brief

A quarterly business report needs to: read professionally, with real navigable structure; get reviewed by a colleague before finalizing; go out personalized to several named recipients; and be summarized into a short, well-designed slide deck for a live stakeholder meeting — plus a version stakeholders who couldn't attend can review afterward.

Step 1 — Structuring the Report with Styles

Every section title uses a real Heading style (Chapter 2) — Heading 1 for major sections, Heading 2 for subsections — never manually bolded text made to look the part. Opening the Navigation Pane confirms the entire report's structure is genuinely visible and jumpable, not just visually convincing.

Step 2 — Page Layout for a Professional Report

Section breaks (Chapter 3) divide the report into a cover page, a Table of Contents section, the main content, and — for one wide data table that doesn't fit portrait width — a single Landscape appendix page, isolated between its own pair of section breaks exactly as Chapter 3's own exercise demonstrated.

Step 3 — Automatic ToC, Captions & Cross-References

A real Table of Contents (Chapter 4) is inserted, built entirely from Step 1's Heading styles. Every chart and table in the report gets a real Insert Caption label, and any in-text reference to one ("see Figure 3") uses a genuine cross-reference field — both staying correct automatically if a figure is later added or reordered.

Step 4 — Headers, Footers & Page Numbers

The cover page uses Different First Page (Chapter 5) to show no header at all. The Table of Contents section is numbered in roman numerals; the main content section's page numbering restarts cleanly at Arabic "1" once it begins. Link to Previous is turned off wherever the header needs to show that section's own heading rather than mirroring the cover page.

Step 5 — Collaborative Review with Track Changes

Before finalizing, a colleague reviews the draft with Track Changes (Chapter 6) switched on the entire time, plus a Comment flagging a genuine open question about one figure rather than silently deleting it. Every suggested edit is reviewed individually — accepted, rejected, or discussed — rather than blindly Accept All'd.

Step 6 — Personalizing the Report via Mail Merge

A short personalized cover letter accompanying the report is built as a Mail Merge (Chapter 7), connected to an Excel Table of recipient names and account details. Match Fields confirms every merge field genuinely lines up with the Excel source's real column headers before running the full merge — not assumed to match just because the names look similar.

Step 7 — Building the Companion Deck with the Slide Master

The PowerPoint deck uses a Slide Master (Chapter 8) whose Theme colours deliberately match the Word report's own accent colour — the same shared-Theme idea Chapter 1 introduced. Every slide uses one of a small handful of defined Layouts, never one-off manually-positioned text boxes.

Step 8 — Distilling the Report into Slides

The deck does not copy the report's own paragraphs onto slides. Following Chapter 9, each slide covers exactly one takeaway from the report, kept within the 6×6 guideline; the report's own dense data table becomes a single clear chart on its own slide instead of a wall of numbers, chosen by the same "what question is this answering" logic Excel Fundamentals Chapter 8 established.

Step 9 — Purposeful Motion

One consistent, simple transition style is applied to every slide (Chapter 10). The single slide walking through several sequential findings uses an On Click Entrance animation so each point appears exactly when the presenter is ready to discuss it — the deck's only animation, used because it solves a genuine pacing need, not decoration.

Step 10 — Rehearsing and Presenting

Every slide's extra explanatory detail — the specifics trimmed out during Step 8's distillation — lives in that slide's own Speaker Notes (Chapter 11), not back on the slide itself. Rehearse Timings confirms realistic pacing before the live meeting. Afterward, the deck is exported both as a video (for stakeholders who couldn't attend live) and as PDF handouts (a quick static reference for anyone wanting to skim it before a follow-up).

Putting It All Together

CAPSTONE PIECE	TECHNIQUE USED	COVERED IN
Report section structure	Real Heading styles, not direct formatting	Chapter 2
Cover page, ToC section, landscape appendix	Section breaks	Chapter 3
Navigable ToC, numbered figures/tables	Automatic ToC, captions, cross-references	Chapter 4
Cover page with no header, roman/Arabic page numbers	Different First Page, Link to Previous, page-number restart	Chapter 5
Colleague review before finalizing	Track Changes and Comments	Chapter 6
Personalized cover letters	Mail Merge from an Excel Table	Chapter 7
A visually consistent deck	Slide Master and Layouts	Chapter 8
One takeaway per slide, a chart over a data table	Content density and visual hierarchy	Chapter 9
One purposeful animated reveal	On Click entrance animation, restrained transitions	Chapter 10
Extra detail kept off the visible slide	Speaker Notes, Presenter View, exported video/handouts	Chapter 11

TWO APPS, ONE SHARED DISCIPLINE

Nearly every step in this capstone traces back to the same underlying idea, applied twice: define formatting centrally and apply it consistently (Styles in Word, the Slide Master in PowerPoint); keep the audience-facing content lean and let a separate mechanism hold the extra detail (Word's own structure and a real leave-behind document, versus PowerPoint's Speaker Notes). Recognising that shared discipline is what actually carries across from one app to the other — not memorising each app's own menu locations.

A FINISHED-LOOKING DELIVERABLE STILL NEEDS ITS OWN FINAL CHECKS

Before this report and deck are actually sent or presented: every field in the Word report (ToC, captions, cross-references) needs a full Update per Chapter 4's own warning, every tracked change from Step 5 needs to be genuinely resolved rather than left pending, the Mail Merge needs a spot-check across several rows (not just the first preview) per Chapter 7, and Presenter View needs testing on the actual presentation equipment per Chapter 11 — none of these are optional final touches, they're the difference between "looks finished" and "actually is."

Hands-On Exercises

EXERCISE 1

Build the Word report structure from Steps 1-4: at least 3 Heading-1 sections, a section break isolating one landscape page with a wide table, an automatic Table of Contents, and a caption on that table. Confirm the Navigation Pane shows your sections and the ToC lists them with correct page numbers.

 [View solution](#)

EXERCISE 2

Using your report from Exercise 1, set up a Mail Merge (Step 6) with a small Excel Table (3-4 rows) of recipient names and one numeric field, generating a personalized cover letter per recipient. Then build a 4-slide companion deck (Steps 7-8) using a Slide Master and one Layout, distilling the report into 4 single-idea slides rather than copying its paragraphs.

 [View solution](#)

EXERCISE 3

A colleague reviewing your capstone project says the deck "looks great" but points out that half the report's key details from the original document are nowhere to be found in the deck at all. Explain why this is actually correct behaviour rather than a mistake, and where that missing detail should genuinely live instead.

 [View solution](#)

COURSE RECAP — WORD & POWERPOINT

- **Ch.1:** two content models under one shared Ribbon/Theme interface — Word's flow, PowerPoint's discrete slides
- **Ch.2:** real Styles vs. direct formatting — the same idea Ch.8's Slide Master later mirrors
- **Ch.3-5:** section breaks, and everything they unlock — orientation, headers, page numbering
- **Ch.4:** automatic ToC, captions, and cross-references, all built from real Heading styles
- **Ch.6:** Track Changes and Comments for genuine multi-person collaboration
- **Ch.7:** Mail Merge, bridging directly to an Excel Table as its data source
- **Ch.8:** the Slide Master — PowerPoint's own version of Ch.2's Styles
- **Ch.9:** content density, visual hierarchy, and choosing a chart over a wall of text
- **Ch.10:** animation and transitions used with restraint, not decoration
- **Ch.11:** Presenter View, Speaker Notes, rehearsing, and exporting for those who couldn't attend
- **Ch.12:** all of the above, combined into one real report-and-deck project