Absolute Beginner's Guide to Technical Authoring

Where to start, who to contact, what to learn, and how to survive in the age of AI

MICHAEL CLARK

About this guide

This guide is for absolute beginners: graduates, apprentices, career changers, subject matter experts who have been asked to write documents, and anyone who has been told, “You are good with words, can you sort the documentation?”

The aim is simple. By the end, the reader should understand what technical authoring is, where to start, who to speak to, what to learn first, and why the role has never been just a writing job.

Core message

Technical authoring is not simply the act of writing. It is the discipline of turning knowledge, evidence, process, risk, decisions and technical detail into information that people can trust and use.

This guide is written from a practical workplace viewpoint. It deliberately avoids academic padding. It focuses on what a new technical author needs to notice, ask, challenge, structure, validate and maintain.

Who this guide is for

- New technical authors and technical writers who do not know where to start.
- Engineering graduates and apprentices who need to write reports, procedures and evidence-based documents.
- People moving from IT support, engineering, service management, quality, training, operations or administration into documentation.
- Managers who think technical authoring means “tidying up words” and need to understand the wider value of the role.
- Experienced writers who want to move beyond writing-only work and protect their value in the age of AI.

Contents

1. What technical authoring really is

A technical author explains complicated things clearly. That is the simple answer, but it is not enough. A good technical author does not merely take technical information and make it sound better. A good technical author finds out what the information is for, who needs it, what decision or task it supports, what evidence proves it, and who will keep it correct after publication.

The word “author” can be misleading. It suggests a person sitting alone writing pages. In reality, most of the work happens before and after the writing. The writing is the visible part. The invisible part is discovery, questioning, planning, interviewing, organising, reviewing, validating, checking, publishing and maintaining.

Beginner rule

Do not ask, “What do I need to write?” first. Ask, “Who needs to use this information, and what must they do with it?”

A practical definition

Technical authoring is the practice of collecting, structuring, writing, reviewing and maintaining information that helps people understand a product, system, service, process, decision, risk or task.

That information may appear as a user guide, procedure, report, policy, service document, API guide, knowledge article, training pack, data centre runbook, disaster recovery plan, SharePoint page, work instruction, audit pack, release note or technical proposal.

The technical author’s real job

Visible task	Hidden work behind it
Write a guide	Understand the audience, purpose, source material, ownership, approval route and maintenance cycle.
Create a procedure	Map the process, confirm hand-offs, identify decisions, capture exceptions, test the steps and

	assign ownership.
Produce a report	Structure the evidence, explain findings, separate fact from opinion, make recommendations and support decisions.
Update a template	Control styles, metadata, document fields, approval language, branding, accessibility and version control.
Build a knowledge article	Define the issue, check the fix, use search terms, apply tags, add screenshots, publish and review.

Why technical authoring matters

Poor documentation is not harmless. It creates operational risk. It makes training harder. It causes repeated mistakes. It slows recovery during incidents. It hides ownership gaps. It makes audits painful. It allows people to argue about what the process is supposed to be because no one has written it down properly.

Good documentation gives an organisation memory. It reduces dependency on individuals. It supports consistency. It shows how decisions were made. It lets new people learn without constantly interrupting experienced staff. It protects the business when someone leaves, a supplier fails, a system breaks, or an auditor asks awkward questions.

Exercise	**Exercise 1: Spot the real task**
Purpose	To separate writing from the wider authoring task.
Task	Take a document you already use at work. List what is visible on the page, then list the hidden work required to keep it accurate.
Output	A two-column list showing visible content and hidden authoring work.

2. The writing roles around technical authoring

Technical authoring overlaps with many other writing and information roles. Job titles vary across sectors. Two roles with the same title can be completely different. Two roles with different titles can ask for almost the same skills. A beginner should therefore read job descriptions carefully and look for the work, not only the title.

Role	Typical focus	What a beginner should notice
Technical Author / Technical Writer	User guides, procedures, manuals, release notes, technical reports, service documents.	Often broader than writing. May include interviews, templates, reviews, document control and publishing.
Documentation Specialist	Document libraries, templates, controlled documents, governance, reviews.	Often strong in process, versioning, metadata and compliance.
Information Manager	How information is organised, owned, stored, found and retained.	Important for SharePoint, document management, records and governance.
Knowledge Manager	Knowledge articles, support information, FAQs, known errors, searchable content.	Closely linked to service desks and operational support.
Content Designer / UX Writer	Interface wording, user journeys, task-based content and digital services.	More focused on user experience and product/service design.
API Writer / Developer Documentation Writer	APIs, SDKs, code examples, developer portals and integration guides.	May require technical depth, code reading and tools such as Git, Markdown or docs-as-code.
Process / Procedure	Business processes, operating procedures	Requires process mapping, SME interviews and

Writer	and work instructions.	validation.
Policy Writer	Policies, standards, governance and compliance documents.	Needs careful language, ownership, approval and evidence.
Bid / Proposal Writer	Tender responses, proposals, service descriptions and capability statements.	More commercial and persuasive, but still evidence-based.
Training Content Developer	Course material, learner guides, exercises, slides and assessments.	Requires learning objectives and practical exercises, not just information dumping.

The mistake beginners make

Beginners often search only for “technical writer” jobs. That can hide better opportunities. Search also for documentation manager, knowledge author, procedure writer, information manager, content designer, service documentation specialist, ITSM documentation analyst, policy analyst, proposal writer and document control specialist.

Practical search tip

When job hunting, search by task as well as title: “procedures”, “runbooks”, “SharePoint”, “knowledge base”, “ITIL”, “service catalogue”, “API documentation”, “controlled documents”, “policy writing” and “documentation governance”.

Writing roles are splitting

The profession is fragmenting. Some roles are becoming code-adjacent. Some are moving towards UX and content design. Some are becoming governance and information management roles. Some sit in service management, project delivery or operational readiness. This is not a bad thing, but it means a beginner needs to choose a direction.

A safe early strategy is to build a broad foundation: plain English, structure, audience analysis, interviews, process mapping, document control, tools and review discipline. After that, specialise.

Exercise	Exercise 2: Role comparison
Purpose	To understand the differences between titles.
Task	Find three job adverts with different titles but similar documentation tasks. Highlight the verbs: write, manage, review, publish, maintain, interview, analyse, configure, govern.
Output	A short comparison showing which role is writing-heavy, tool-heavy, process-heavy or governance-heavy.

3. Myths that damage technical authoring

Many people misunderstand technical authoring. Some think it is a writing service. Some think it is formatting. Some think the subject matter expert can simply “send the facts” and the writer will turn them into a document. These myths cause weak documents, poor planning and unrealistic expectations.

Myth	Reality
Technical authoring is only writing.	Writing is only one part. The role includes discovery, questioning, structuring, validation, ownership, publication and maintenance.
You must be a developer to be a technical author.	Some roles require code knowledge. Many do not. Process, service, governance and operational documentation need analytical thinking and good questioning.
The subject matter expert knows what the reader needs.	SMEs know the subject. They do not always know the reader, the task, the level of detail or the document structure.
A template fixes documentation problems.	A template helps consistency. It cannot fix weak ownership, missing facts, poor review or unclear purpose.
AI will write all documentation.	AI can draft text, summarise and assist. It cannot own accuracy, accountability, approvals, evidence, operational risk or organisational consequences.
A good writer can write anything without help.	Technical authors need access to SMEs, systems, users, evidence and review cycles. Writing without source access is guesswork.
More detail means better documentation.	Too much detail can bury the task. Useful documentation has the right detail for the right reader at the right time.
Once a document is published, it is finished.	Publication is a point in the lifecycle. Documents need review, ownership, updates, retirement and archive rules.

The most dangerous myth

The most dangerous myth is that the technical author is only responsible for wording. If the role is reduced to wording, the role becomes easier to automate, outsource or ignore. If the role is understood as information control, risk reduction, process clarity and user enablement, it becomes much harder to dismiss.

Field truth

If you want to survive, do not sell yourself only as a writer. Sell yourself as someone who can make information usable, controlled, auditable and maintainable.

Exercise	Exercise 3: Myth buster
Purpose	To practise explaining the value of technical authoring.
Task	Choose three myths from the table and write a short reply to a manager who believes each one.
Output	Three short professional responses, each no more than 80 words.

4. Where a beginner should start

A beginner does not need to know everything on day one. The important thing is to build the right foundation. Technical authoring rewards curiosity, structure, patience, scepticism and the ability to ask simple questions without sounding foolish.

Start with the fundamentals

- Audience: who will read this and what do they already know?
- Purpose: what does the document help the reader do, decide, understand or prove?
- Scope: what is included and what is deliberately excluded?
- Source: where does the information come from?
- Owner: who is accountable for the content after publication?
- Review: who must check accuracy, compliance and usability?
- Format: what type of document is needed?
- Lifecycle: how will the document be maintained, reviewed, archived or retired?

The first skills to learn

Skill	Why it matters	Beginner practice
Plain English	Readers need clear instructions, not impressive wording.	Rewrite long sentences into short, active sentences.
Audience analysis	Different readers need different levels of detail.	Write the same instruction for a novice and an expert.
Structure	A badly structured document is hard to use even if the sentences are clear.	Create outlines before drafting.
Interviewing SMEs	Most technical information lives in people's heads.	Prepare questions before meetings and confirm answers afterwards.

Skill	Why it matters	Beginner practice
Process mapping	Procedures depend on sequence, decisions and hand-offs.	Draw a simple flow before writing steps.
Review discipline	A document is not reliable until it has been checked.	Use a review checklist rather than asking for vague comments.
Tool control	Poor Word, SharePoint or Confluence habits create chaos.	Learn styles, metadata, versioning and permissions.
Evidence thinking	Reports and audits need proof, not opinion.	Separate fact, assumption, risk and recommendation.

Tools worth learning early

Do not try to learn every tool. Start with the tools most employers use. Build confidence in the boring tools first because most workplace documentation still depends on them.

- Microsoft Word: styles, headings, tables, captions, cross-references, templates and track changes.
- PowerPoint: clear diagrams, training material and stakeholder presentations.
- Excel: registers, document inventories, metadata lists and simple dashboards.
- SharePoint: libraries, metadata, version control, permissions and document lifecycle.
- Visio or draw.io: process maps, system diagrams and swimlanes.
- Confluence: team knowledge bases, page structures, labels and templates.
- Markdown and Git basics: useful for docs-as-code and developer documentation roles.
- AI tools: useful for drafting and analysis, but only when source control and validation are understood.

Do not begin with tools alone

Tools help you produce documents. They do not teach you what the document is for. Always learn purpose, audience and structure before worrying about clever formatting.

Exercise	Exercise 4: First document assessment
Purpose	To learn how to judge a document before rewriting it.
Task	Choose one existing document. Identify the audience, purpose, owner, review date, source material, decision points and missing information.
Output	A one-page document assessment sheet.

5. Who to contact and what to ask

Technical authoring is a contact sport. You cannot produce reliable documentation by sitting quietly and hoping the truth arrives by email. You need to find the people who know the work, the people who own the work, the people who use the output and the people who approve the result.

The people you normally need

Contact	Why they matter	Questions to ask
Service owner / product owner	Accountable for the service or product.	What is this for? Who uses it? What must not go wrong?
Subject matter expert	Knows the technical detail.	What are the steps? What exceptions occur? What do people usually misunderstand?
Operations / support team	Knows what happens in real life.	What tickets repeat? What fails? What workaround is used?
End users	Know whether the document is usable.	Where do you get stuck? What words do you search for? What do you need first?
Security / risk / compliance	Knows legal, audit or security constraints.	What must be controlled? What evidence is required? What must not be published?
Change manager	Knows release, change and approval rules.	What changed? What documents must be updated before go-live?
Document control / information manager	Knows naming, metadata, versioning and storage rules.	Where will this live? What metadata is mandatory? What review cycle applies?
Project manager	Knows timescales, dependencies and stakeholders.	What is the deadline? What decisions are pending? Who signs off?
Vendor / supplier	May own system	What are the official steps?

Contact	Why they matter	Questions to ask
	knowledge or support obligations.	What is contractually supported? What is excluded?

Questions beginners should ask without apology

- What problem is this document solving?
- Who is the primary reader?
- What will the reader do after reading it?
- What level of knowledge can we assume?
- What source material already exists?
- Which parts are confirmed fact and which are assumptions?
- What must be approved before publication?
- What happens if this document is wrong?
- Who will maintain it after publication?
- When should it be reviewed or retired?

Meeting discipline

Never leave a discovery meeting with only “useful context”. Leave with actions, owners, missing facts, source documents, review names and a clear next step.

Exercise	**Exercise 5: SME interview plan**
Purpose	To prepare for technical discovery.
Task	Pick a technical topic and create ten questions for the SME. Include at least two questions about exceptions, two about risks, and two about ownership.
Output	A structured SME interview sheet.

6. The practical documentation workflow

The beginner’s temptation is to start writing too soon. Resist it. The fastest way to waste time is to draft a document before the purpose, audience, structure and source material are understood.

A simple seven-stage workflow

Stage	What happens	Output
1. Define	Agree purpose, audience, scope, owner and document type.	Document brief.
2. Discover	Collect source material, interview SMEs, review systems and identify gaps.	Source pack and question log.
3. Structure	Create outline, headings, flow, tables, diagrams and appendix plan.	Outline prototype.
4. Draft	Write the first controlled draft using plain English and agreed structure.	Draft 0.1.
5. Validate	Check technical accuracy with SMEs and usability with representative readers.	Reviewed draft and action log.
6. Approve and publish	Complete sign-off, metadata, versioning, accessibility and storage.	Published document.
7. Maintain	Review, update, retire, archive and measure use.	Controlled lifecycle.

What not to do

- Do not accept “just tidy this up” without asking who owns the content.
- Do not write from memory when source material exists.
- Do not let reviewers give vague comments such as “needs more detail” without explaining where and why.
- Do not publish a document that has no owner, no version, no review date and no storage location.
- Do not confuse a procedure with a policy. A policy says what must be true; a procedure says how the work is done.

- Do not create deep folder structures when metadata, views and libraries would make retrieval easier.
- Do not make AI-generated text look authoritative unless it has been checked against reliable sources.

Minimum document control fields

Field	Why it matters
Document title	Makes the document identifiable and searchable.
Document type	Shows whether it is a policy, procedure, work instruction, report, guide or checklist.
Owner	Names the person or role accountable for content.
Author	Shows who prepared the document.
Reviewer	Shows who checked accuracy.
Approver	Shows who authorised publication.
Version	Controls change history.
Status	Draft, review, approved, published, archived.
Last updated date	Shows currency.
Next review date	Prevents stale documentation.
Location	Shows the controlled source of truth.

Exercise	**Exercise 6: Build a document brief**
Purpose	To stop uncontrolled drafting.
Task	Create a one-page brief for a procedure, report or user guide before writing any content.
Output	A document brief containing purpose, audience, scope, owner, source material, reviewers and deadline.

7. What a technical author needs to learn

Technical authoring is a layered role. The beginner starts with writing and structure, then moves into process, tools, quality, governance and risk. The more layers you can handle, the more resilient your career becomes.

Layer	What it means	Why it protects the role
Writing craft	Plain English, grammar, headings, paragraphs, lists and tone.	Still essential, but not enough on its own.
Technical understanding	Enough knowledge to ask useful questions and spot weak answers.	Prevents blind transcription from SMEs or AI.
Process thinking	Understanding sequences, decisions, hand-offs and exceptions.	Supports procedures, work instructions and ITSM documentation.
Information architecture	Organising content so it can be found, reused and maintained.	Useful in SharePoint, Confluence, knowledge bases and content systems.
Governance	Ownership, review, approval, versioning and evidence.	Moves the role beyond writing into control and assurance.
User research	Understanding what readers actually need.	Prevents documents becoming internal vanity projects.
Tool capability	Word, SharePoint, Confluence, Markdown, diagrams and AI tools.	Makes the author productive and credible.
AI literacy	Prompting, source checking, validation, risk and limitations.	Allows the author to use AI without becoming dependent on it.

Useful subject areas

- ITIL and ITSM: incident, change, problem, service request, service catalogue, supplier management and knowledge management.
- Information management: libraries, metadata, ownership, retention, archive and search.
- Quality management: review, audit trails, non-conformance, corrective action and continual improvement.
- Security and compliance: access, confidentiality, data protection, evidence and approvals.
- Business continuity and disaster recovery: recovery procedures, contacts, RTO, RPO and testing evidence.
- Engineering communication: reports, calculations, diagrams, assumptions, design evidence and safety information.
- Project delivery: requirements, handover, operational acceptance and lessons learned.

Certifications and learning routes

A beginner does not need a wall full of certificates. However, structured learning can help. Consider basic courses in technical writing, plain English, ITIL Foundation, SharePoint administration, information management, business analysis, quality management or service management. Choose courses that support the type of role you want.

Do not collect qualifications at random. Link each one to a workplace skill: writing, process, service management, information management, audit, accessibility, user research, content design or AI governance.

Exercise	Exercise 7: Skill map
Purpose	To plan your own development.
Task	Create a table with three columns: skill I have, skill I need, proof I can show. Include at least one writing skill, one process skill, one tool skill and one governance skill.
Output	A personal technical authoring skill map.

8. Technical writing and report writing basics

Technical writing and report writing overlap, but they are not identical. Technical writing often helps someone perform a task or understand a system. Report writing usually helps someone understand findings, evidence and recommendations. Both need clarity, structure and discipline.

Technical writing basics

- Write for the reader, not for the person who gave you the information.
- Use active voice where possible: “Restart the service” is clearer than “The service should be restarted”.
- Use consistent terms. Do not call the same thing a system, platform, tool and application without reason.
- Break procedures into numbered steps.
- Use warnings before the step where harm or failure could occur.
- Use tables for comparison and decision information.
- Use diagrams where relationships, flows or architecture are easier to see than describe.
- Keep paragraphs short and focused.
- Define acronyms the first time they appear.
- Do not include clever wording if simple wording works.

Report writing basics

Report section	Purpose
Executive summary	Gives busy readers the issue, finding, implication and recommendation.
Background	Explains why the report exists.
Scope	Defines what was and was not examined.

Report section	Purpose
Method	Explains how evidence was gathered.
Findings	Presents facts and evidence.
Analysis	Explains what the findings mean.
Risks or implications	Shows what may happen if no action is taken.
Recommendations	States what should happen next.
Action plan	Names owners, dates and next steps.
Appendices	Holds supporting detail without cluttering the main argument.

The difference between fact, assumption and opinion

Beginners often blur fact, assumption and opinion. This matters because technical documents may influence decisions, budgets, incidents, safety, service recovery and audit outcomes.

Type	Example	How to handle it
Fact	The server was restarted at 14:10.	Support with evidence where needed.
Assumption	The outage may have been caused by a failed dependency.	Label it clearly and confirm later.
Opinion	The system is badly designed.	Avoid unless supported by evidence and framed professionally.
Recommendation	Implement monitoring for the failed dependency.	Link to evidence, risk and owner.

Exercise	Exercise 8: Rewrite for clarity
Purpose	To practise plain technical writing.
Task	Rewrite this sentence: “It is considered necessary that the appropriate stakeholders are engaged in order that the documentation can be progressed in a timely manner.”
Output	A clear sentence of fewer than 15 words.

Exercise	Exercise 9: Report finding
Purpose	To practise evidence-based reporting.
Task	Write one finding, one implication and one recommendation for a repeated service outage.
Output	Three short paragraphs: Finding, Implication, Recommendation.

9. AI: threat, tool or turning point?

AI is not a side issue for technical authors. It directly affects writing, summarising, drafting, reviewing and knowledge retrieval. It can make a good technical author faster. It can also make a weak organisation believe it no longer needs one.

The realistic position is neither panic nor complacency. AI will reduce the value of simple, unsupervised text production. It will increase the value of people who can validate information, control sources, manage ownership, structure knowledge, challenge errors and understand the operational consequences of wrong documentation.

What AI can help with

- Creating first-draft outlines.
- Summarising long source notes.
- Rewriting dense text into plainer language.
- Generating checklists from an agreed process.
- Comparing two versions of content.
- Suggesting headings and structure.
- Finding gaps in a procedure if given the right context.
- Creating training questions from approved content.

What AI must not be trusted to own

- Technical accuracy without source verification.
- Legal, safety, security or regulatory claims.
- Final approval of operational procedures.
- Document ownership or accountability.
- Current system behaviour unless connected to verified current sources.
- Confidential information handling unless the organisation has approved the tool and data controls.
- Decisions about what the business is prepared to risk.

AI principle

AI can help produce words. It cannot attend the incident review, explain the failed recovery step, justify missing approval, or accept responsibility for wrong instructions.

A safe AI workflow for technical authors

1. Start with controlled source material.
2. Tell the tool the audience, purpose, format and boundaries.
3. Use AI to draft or restructure, not to invent facts.
4. Check every technical claim against source material or an SME.
5. Mark unknowns and assumptions clearly.
6. Keep the human review and approval route intact.
7. Record where AI was used if the organisation requires disclosure.
8. Never paste sensitive, confidential or restricted material into an unapproved tool.

What current evidence suggests

Recent labour-market evidence does not support a simple “all writers vanish” story. It does, however, support a serious shift in skills. The World Economic Forum’s 2025 Future of Jobs reporting projected major disruption by 2030, with large numbers of jobs both created and displaced and nearly 40% of job skills expected to change. The U.S. Bureau of Labor Statistics notes that technical writer employment growth may be slowed by AI tools that make writers more productive. UK government AI adoption research also describes AI as a support layer that can redefine roles by moving human staff towards more interactive, creative or strategic work. The practical conclusion is clear: writing-only work is exposed; accountable information work is more defensible.

Exercise	**Exercise 10: AI validation test**
Purpose	To learn how to use AI safely.
Task	Ask an AI tool to draft a short procedure from approved notes. Then mark every sentence as verified, needs checking, assumption or remove.
Output	A marked-up draft showing where human validation is required.

10. Can technical authoring roles survive?

Yes, but not unchanged. It is realistic to expect technical authoring roles to survive where the role moves beyond simple writing. It is not realistic to expect every traditional writing-only task to remain protected. Organisations will use AI for first drafts, summaries, translation, rewriting and content conversion. Some will use it well. Some will use it badly. The technical author who can only produce clean prose will be under pressure.

The technical author who understands users, systems, process, information architecture, governance, evidence, service management, review cycles and AI validation has a stronger future. That person is not merely competing with AI. That person is controlling how information is produced, tested and trusted.

Roles likely to be under pressure

- Pure rewrite roles with little subject access.
- Low-value formatting-only roles.
- Documentation jobs where no one expects source validation.
- Routine content conversion without analysis.
- Generic content writing where accuracy is not strongly governed.

Roles likely to remain valuable

- Technical authors who can interview SMEs and challenge gaps.
- Documentation managers who control ownership, review and publication.
- Information managers who design usable repositories and metadata.
- ITSM documentation specialists who understand incident, change, problem, service catalogue and knowledge management.

- Operational readiness authors who make sure services are supportable before go-live.
- Knowledge managers who keep support content accurate and searchable.
- Technical authors who can use AI responsibly and validate its output.
- Writers who understand risk, audit, governance, data protection and service consequences.

Realistic answer

The role survives if it evolves. The job title may change. The writing may become faster. The value will move towards judgement, validation, structure, ownership and information control.

The future titles to watch

Possible future title	What it may involve
Information Consultant	Advising on document ecosystems, governance, structure and ownership.
Documentation Governance Lead	Controlling standards, reviews, approvals, quality and lifecycle.
Knowledge Systems Lead	Managing knowledge bases, article quality, search and support content.
AI Documentation Validator	Checking AI-assisted documentation for accuracy, completeness and traceability.
Service Documentation Manager	Owning ITSM and operational documentation across live services.
Content Operations Specialist	Managing workflows, publication pipelines, metadata and content quality.
Technical Information Architect	Designing the structure and relationships of technical content.
Operational Readiness Documentation Lead	Ensuring projects hand over usable support, DR, monitoring and process documentation.

Exercise	**Exercise 11: Future-proof your role**
Purpose	To identify where your value currently sits.
Task	Rate yourself from 1 to 5 against writing, SME interviewing, process mapping, tool control, governance, AI use, service knowledge and information management.
Output	A personal future-readiness scorecard and three development actions.

11. Ninety-day beginner development plan

A beginner needs direction. The following plan is practical and achievable. It is not about becoming an expert in three months. It is about building enough foundation to become useful, credible and employable.

Period	Focus	Actions	Evidence to produce
Days 1-30	Foundation	Learn plain English, document types, audience analysis, Word styles, basic SharePoint, and technical authoring workflow.	One rewritten procedure, one document assessment, one glossary.
Days 31-60	Discovery and structure	Practise SME interviews, process mapping, outlining, review checklists and document control.	One interview sheet, one process map, one document brief, one review checklist.
Days 61-90	Portfolio and specialism	Choose a direction: ITSM, engineering, API, IM, knowledge, policy or procedures. Build samples and practise AI validation.	Three portfolio samples and a one-page career direction statement.

A beginner portfolio

You do not need client secrets or confidential material to build a portfolio. Create sample documents based on imaginary systems, public information or your own scenarios. The point is to show method, structure and judgement.

- A one-page quick start guide.
- A short procedure with numbered steps and warnings.

- A work instruction with screenshots or placeholders.
- A process map with roles and hand-offs.
- A technical report with findings and recommendations.
- A SharePoint library metadata design.
- A knowledge article and known error record.
- A document control checklist.
- An AI-assisted draft with a validation log.

Where to find learning and contacts

- Internal SMEs: engineers, developers, support teams, service owners and project managers.
- Professional communities: technical communication groups, ITSM groups, content design communities and LinkedIn discussions.
- Workplace projects: migrations, process clean-ups, service transition, template fixes, audit preparation and knowledge base work.
- Training routes: technical writing, plain English, ITIL Foundation, SharePoint, business analysis, information management, content design and AI literacy.
- Mentors: experienced technical authors, documentation managers, service managers, quality managers and information managers.

Exercise	**Exercise 12: Build your first portfolio piece**
Purpose	To create proof of ability.
Task	Invent a simple system such as an access request portal. Create a quick start guide, a support procedure and a knowledge article for it.
Output	Three short documents showing audience, structure, steps, warnings and metadata.

12. Exercises, templates and checklists

This section collects practical tools that a beginner can use immediately. They are deliberately simple. Complex templates can come later. The beginner first needs discipline.

Template: Document brief

Field	Complete before drafting
Working title	
Document type	
Purpose	
Primary audience	
Secondary audience	
In scope	
Out of scope	
Owner	
Author	
SMEs	
Reviewers	
Approver	
Source material	
Risks if wrong	
Target publication date	
Review date	

Template: SME interview log

Question	Answer	Owner / follow-up
What is the purpose of the document?		
Who uses the information?		
What are the normal steps?		
What exceptions occur?		
What must the user never do?		
What evidence proves this is correct?		
What systems, teams or suppliers are involved?		
Who approves the final document?		

Checklist: Before drafting

- Purpose agreed.
- Audience agreed.
- Document type agreed.
- Owner named.
- Source material collected.
- SMEs identified.
- Reviewers identified.
- Approver identified.
- Publication location agreed.

- Version and review rules understood.

Checklist: Before publication

- The document has a clear title.
- The owner is named by role or person.
- The version number is correct.
- The status is correct.
- The last updated date is correct.
- The next review date is set.
- Technical accuracy has been checked.
- The intended reader can follow the document.
- Acronyms and technical terms are defined.
- Warnings and critical steps are visible.
- Links work.
- Screenshots are current or deliberately omitted.
- The document is stored in the controlled location.
- Old versions are archived or clearly superseded.

Checklist: AI-assisted documentation

- The source material is approved for use.
- No restricted or confidential information has been pasted into an unapproved tool.
- AI was used for drafting, structuring or rewriting, not unsupported fact creation.
- Every technical claim has been checked.
- Assumptions have been marked.
- SME review has not been skipped.
- The final document has a human owner and approver.
- The document can be defended if challenged.

Quick glossary

Term	Meaning
Audience	The people who need to use the information.

Term	Meaning
Content owner	The person or role accountable for keeping content accurate.
Document control	The rules for versioning, approval, publication and review.
Information architecture	The structure that makes information findable and usable.
Metadata	Information about information, such as owner, type, status, review date and service.
Procedure	A sequence of steps showing how to complete a task or process.
Process	A flow of activities, decisions and hand-offs that produces an outcome.
SME	Subject matter expert. The person who knows the technical or operational detail.
Technical author	A person who structures, writes and manages technical information for users, teams or organisations.
Work instruction	A detailed instruction for performing a specific task.

Closing argument: do not remain only a writer

A beginner may enter technical authoring because they can write. That is fine. Writing is still important. Clear sentences matter. Good structure matters. Plain English matters. But if you want a durable career, do not stop there.

Learn how organisations actually work. Learn how services are supported. Learn how documents are controlled. Learn how incidents happen. Learn how changes are approved. Learn why disaster recovery plans fail. Learn how SharePoint becomes a dumping ground. Learn how SMEs think. Learn how users search. Learn how auditors ask questions. Learn how AI can help and how it can mislead.

The safest technical author is not the person who writes the prettiest paragraph. It is the person who can make complex information clear, usable, owned, reviewed, controlled and trusted.

Final thought

The future technical author is part writer, part analyst, part interviewer, part information manager, part quality controller and part risk spotter. That is why the role can survive, but only if the role is allowed to grow.

Source note and further reading

The market observations in this guide are informed by current public sources on AI, jobs and service management. They are not predictions of any single employer's behaviour.

- World Economic Forum, Future of Jobs Report 2025: projected job disruption, job creation/displacement and changing skill requirements by 2030.
- U.S. Bureau of Labor Statistics, Occupational Outlook Handbook: Technical Writers, including comments on slower growth and AI-enabled productivity.
- UK Department for Science, Innovation and Technology, AI Adoption Research, including evidence that AI is often used as a support layer that can redefine roles.
- PeopleCert, ITIL 4 practice guidance, including emphasis on roles, responsibilities, automation, tools, knowledge, communication, practices and value streams.