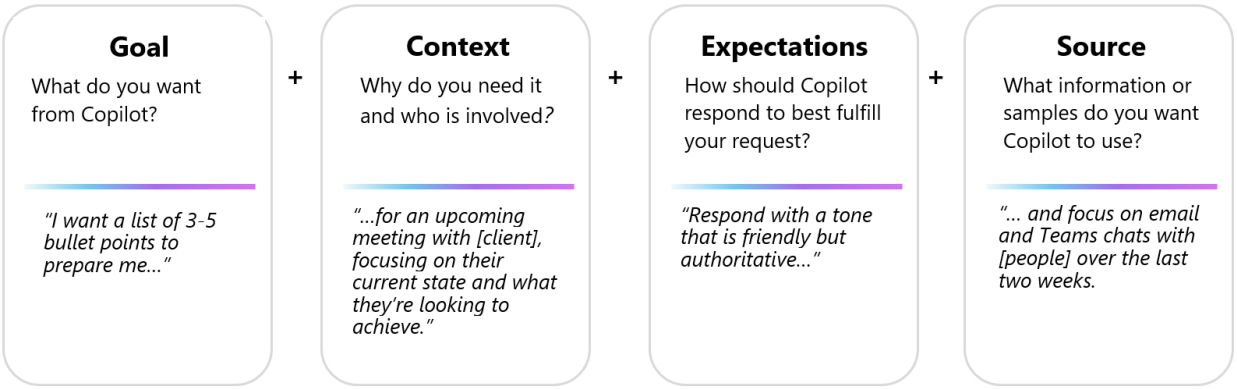
PROMPT 101 Guide

A Framework for Effective Copilot Prompts

Understanding Prompt Architecture



Copilot prompts are the directives or inquiries used to command the AI. An effective prompt can be structured around four core components:

- **Goal:** The specific task you want Copilot to perform.
- **Context:** The background, purpose, or environment for the task.
- **Expectations:** The desired format, tone, or structure of the output.
- **Source:** The specific information or data set you want Copilot to use.

While a clear goal is the only mandatory component, providing additional components (Context, Expectations, Source) will significantly refine the quality and relevance of the results.

Example 1: (Goal + Source)

"Write a summary based on all emails from Sam in the past two weeks."

Example 2: (Goal + Context + Expectations)

"Draft an outline of a training manual about time management. The intended audience is professionals in a hybrid work environment who manage virtual meetings and deadlines. The tone should be friendly and suggestive."

Effective use of Copilot often involves an iterative, conversational process. Expect to refine the initial results with follow-up prompts to achieve your desired outcome.

Copilot Capabilities and Applications

Copilot is built upon Large Language Models (LLMs) integrated with Microsoft 365 applications and university data. This allows Copilot to assist with tasks by securely accessing internal resources such as reports, emails, presentations, and other documents.

Key functional capabilities include:

Information Synthesis

You can use Copilot to synthesize information from various sources.

- **Example (Teams):** "Summarize key questions and action items from this meeting."
- **Example (General):** "What were the main outcomes of Project X as detailed in the latest report?"

Content Generation

Copilot can generate foundational drafts for various communications and documents.

- **Example (PowerPoint):** "Generate a foundational presentation on time management principles."
- **Example (Outlook):** "Draft a professional email congratulating the project lead and team on their successful launch."

Inquiry and Ideation

Use Copilot as a tool for brainstorming and initial planning.

- **Example (Planning):** "Propose three potential team-building activities suitable for a hybrid department."
- **Example (Strategy):** "What are some common frameworks for managing a new project initiative?"

Content Refinement

Copilot can be used to edit, rewrite, or enhance existing content.

- **Example (Word):** Select a paragraph and use the "Rewrite with Copilot" function to improve clarity or conciseness.
- **Example (PowerPoint):** "Incorporate a relevant image, such as a target with arrows, to enhance this slide."

Capabilities for University-Licensed Users

For staff with enterprise licenses, Copilot unlocks enhanced value by connecting LLMs directly to internal university data. This enables more complex, context-aware prompts:

- "Create a training course outline to onboard new staff to Project X."

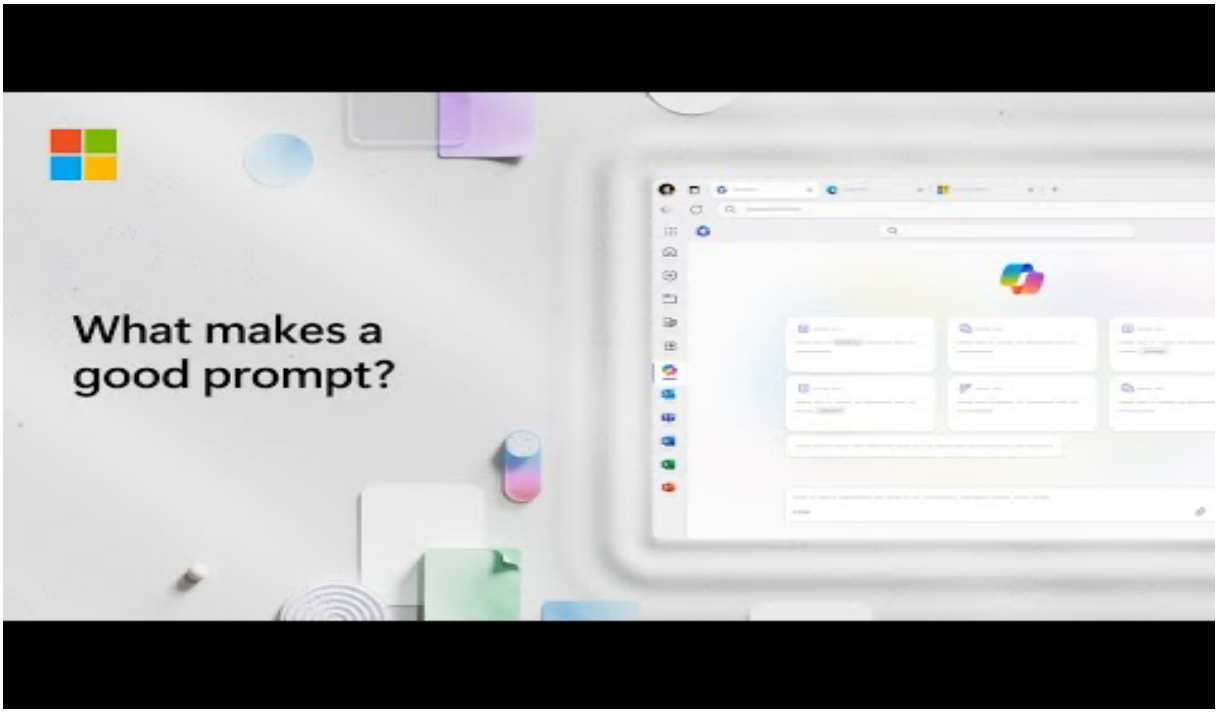
- "Summarize recent communications from [Colleague Name]."
- "Generate a project kick-off presentation based on the topics discussed in the attached chat log."

Important Considerations and Best Practices

1. **Review and Verify Responses:** Copilot is a tool designed to predict and generate text. On occasion, responses may contain inaccuracies. All output should be evaluated for accuracy and cross-referenced with authoritative sources as needed.
2. **Response Variability:** The underlying models introduce a degree of randomness. Using the same prompt multiple times may result in slightly different responses. This is an expected behavior of the technology.
3. **Adherence to University AI Principles:** This tool must be used in a respectful, ethical, and lawful manner. Avoid using Copilot for any purpose that may cause harm. All use must align with CU Denver's responsible AI principles and data governance standards.

Here is that text revised to match the authoritative, professional, and resource-driven tone of a university guide.

A Framework for Effective Copilot Prompting



Backup Link: https://support.microsoft.com/en-us/topic/get-better-results-with-copilot-prompting-77251d6c-e162-479d-b398-9e46cf73da55

Optimizing Outcomes Through Prompt Engineering

The quality of the output from Copilot is directly correlated with the precision and clarity of the prompt. Effective prompt engineering is a key competency for leveraging this technology successfully.

This guide outlines several key methodologies to enhance the quality, relevance, and accuracy of Copilot's responses, enabling staff to optimize their workflows.

1. Employ Parameter-Rich Prompts

A prompt's effectiveness increases significantly with the inclusion of specific parameters. Beyond stating the **Goal** (the desired action), supplement the prompt with:

- **Context:** The operational background, purpose, or intended audience.
- **Expectations:** The required format, tone, length, or structure.
- **Source:** Specific data sets, files, or communications to be used as a reference.

For example, a low-parameter prompt like *`Prompt 1: Write a blog post about sustainable practices in agriculture`* will yield a generic response.

In contrast, a parameter-rich directive like *`Prompt 2: Craft a 1500-word blog post...`* provides comprehensive parameters for audience, scope, and content. This will result in a highly-targeted and more useful first draft.

2. Optimize Instructional Order

The sequence of instructions within a prompt can influence the final output, as the model may place greater emphasis on later instructions.

We recommend experimenting with prompt structure to understand this impact. If directing Copilot to use specific files or sources, it is often best practice to place this directive at the end of the prompt to ensure it is prioritized.

For example, compare *`Prompt 1 (instruction-context-example)`* with *`Prompt 2 (context-example-instruction)`*. The second prompt, which provides foundational context *before* the primary instruction, often produces a more informed and well-grounded response.

3. Utilize Affirmative Directives

Copilot is engineered to take action. Therefore, prompts should be constructed using affirmative instructions ("Do X") rather than negative constraints ("Do not do Y"). This "positive instruction" approach is more effective and reliable. For complex conditional tasks, employ "if-then" logic to guide the output.

4. Engage in Iterative Refinement

The initial response from Copilot should be viewed as a foundational draft, not a final product. Effective outcomes are almost always achieved through an iterative process.

If a response does not meet requirements, refine the prompt with greater specificity and regenerate. The three "Round" examples provided in the source text demonstrate this concept well: a prompt evolves from

a general request (*Round 1*) to a detailed, professional brief (*Round 3*), with each step dramatically improving the quality of the output.

5. Conduct Mandatory Review and Validation

All content generated by Copilot requires critical review and verification by the user.

As Large Language Models (LLMs), these tools are designed to predict and generate text. This process can occasionally lead to inaccuracies, biases, or "hallucinations" (incorrect content). It is the professional responsibility of the staff member to validate all information for accuracy and appropriateness before use or dissemination.

Customizing Copilot Prompts for Specific Use Cases

Adapting Prompts to Meet Your Requirements

Copilot prompts are the directives or inquiries used to command the AI. While the Copilot Prompt Gallery provides numerous examples, these are intended as foundational templates. They should be modified to suit your specific needs.

This article describes the methodology for adapting gallery prompts to align with your precise objectives.

How to Modify a Template Prompt

The Prompt Gallery offers templates that can be customized. Some editable fields are explicitly denoted by brackets, such as `[topic]` or `[file]`.

However, effective use requires moving beyond simple replacement. You should refine all core components of the prompt—including the **Goal**, **Context**, **Expectations**, and **Source**—to match your required outcome.

Consider the following example. A user needs to develop a training resource for new staff and selects a related template from the gallery. The user then modifies the original prompt to create a new, customized directive that precisely matches their goal.

Original Template	Customized/Edited Prompt
Create a script to explain `[concept]` to a `[role/target audience]`, including an explanation of what it is, how it works, and a value proposition. Include 2 analogies to help explain it to someone new to this concept.	Create a **course outline** to explain **Search Engine Optimization (SEO)** to **marketing interns**, including an explanation of what it is, how it works, and **best practices**. Include 2 analogies to help explain it to someone new to this concept. **Include 5 quiz questions at the end of the outline.**

A Guide to Effective Prompting: Optimizing Copilot Results

Understanding Prompt Methodology

Prompts serve as the directives you provide to Microsoft 365 Copilot to execute a task—such as creation, summarization, editing, or transformation.

Effective prompting is best approached as a professional conversation. You must provide plain, clear language and relevant context, just as you would when delegating to a professional assistant.

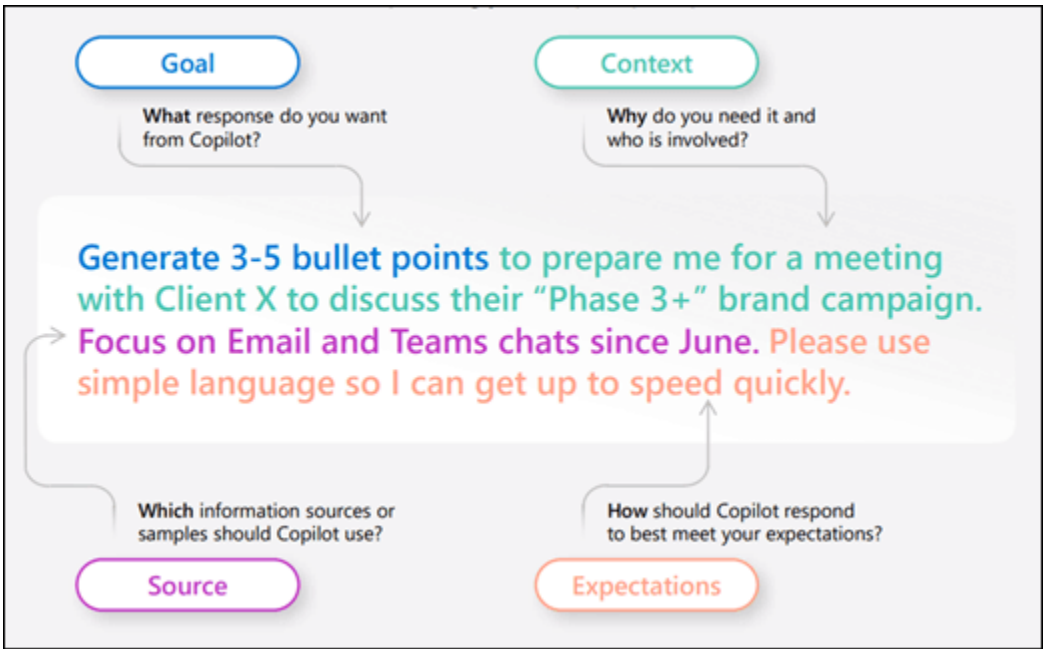
1. Define the Task: Common Use Cases

Your directive should begin with a clear, actionable goal. Common applications for staff include:

- **Summarize Information:** "Write a one-page executive summary of this /[report]."
- **Analyze and Edit Text:** "Review this project rationale for inconsistencies and clarity."
- **Generate Content:** "Create a value proposition for our new staff onboarding initiative."
- **Transform Documents:** "Create an onboarding presentation for new hires based on this /[internal guide]."
- **Track External Developments:** "Give me a concise summary of recent news regarding [new federal grant guidelines]."
- **Receive Project Updates:** "What's the latest on [Project Y], based on recent emails?"

2. Include Key Prompt Ingredients

To achieve optimal responses, it is essential to structure your Copilot prompts with several key elements, as detailed in our other guides.



3. Employ Iterative Refinement (The Conversation)

Engaging in follow-up inquiries allows you to collaborate with Copilot, refining the output to achieve more useful and tailored responses. This iterative process is key to leveraging the tool effectively.

- **Content and Ideation:** Initiate with broader requests, then provide specific parameters to refine the content.
- **Insight and Analysis:** Request a summary of a specific file, then pose relevant, targeted questions to gain deeper insights.
- **Meeting Support:** Request a meeting recap, then ask for clarification on key decisions, action items, or stakeholder sentiment.
- **Technical Problem-Solving:** Present a technical challenge, then iteratively narrow the scope or request step-by-step guidance.

4. Key Principles for Effective Prompting

- **Understand Contextual Limitations:** Copilot's knowledge is limited to the current conversation. Provide comprehensive details and context for each new task.
- **Maintain Professional Language:** Using polite, professional, and clear language improves the quality and alignment of Copilot's responses.
- **Ensure Clarity and Precision:** Pay close attention to punctuation, grammar, and capitalization to avoid ambiguity in your directives.

- **Utilize Quotation Marks:** Employ quotation marks to clearly designate specific text for modification, replacement, or direct analysis.
- **Initiate New Topics Clearly:** To avoid context-blending between tasks, type "new topic" or begin a new chat when switching to an unrelated objective.