

- Identify data that might change and replace it with appropriately named input variables. This allows you to create dynamic prompt templates that can be easily customized for different scenarios.

By following these principles, you can create prompt templates that effectively leverage input parameters to generate dynamic and personalized content.

HANDS-ON PRACTICE

Try creating prompt templates with inputs for the following scenarios:

1. A travel agency wants to generate personalized itineraries for its customers based on their preferences and budget.
2. A fitness app wants to provide customized workout plans for its users based on their fitness level, goals, and available equipment.
3. A recipe website wants to offer personalized recipe suggestions based on a user's dietary preferences and available ingredients.

Focus on identifying the dynamic elements that can be replaced with specific values and creating placeholders for them in your prompt templates. Remember to populate the placeholders with actual values before sending the prompts to the language model for text generation.

3.4 Output Format

Specifying the desired output format is an essential aspect of prompt design. It helps the language model understand how to structure and present the generated content. By providing clear guidelines on the expected output format, you can ensure that the model generates content that is easy to parse, integrate, and utilize in downstream applications. Three common patterns for specifying output format are the Output Indicator Pattern, the Template Pattern, and the Output Style and Tone.

3.4.1 Output Indicator Pattern

The Output Indicator Pattern involves using an indicator to signal the language model where to start generating the output and what format to use. This pattern is useful when you want the model to generate content in a specific format without providing a complete template.

The output indicator primes the language model to generate content in the specified format. For example, if the prefix is 'Full Name:', the language model will generate a full name. Similarly, if the prefix is 'Poem:', the model will attempt to generate a poem.

3.4.2 Practical Example 1: Generating a Product Review

Suppose you want to generate a detailed product review for the SmartX Pro smartphone based on the context provided.

PROMPT

...

Generate a detailed product review for the SmartX Pro smartphone Product Review.

Product Review:

Pros:

- Stunning OLED display with vibrant colors and deep blacks*
- Powerful octa-core processor for smooth performance*
- Long-lasting battery life with fast charging support*

Cons:

- Slightly expensive compared to similar smartphones*
- Camera performance could be better in low-light conditions*

Overall Rating: 4/5

Detailed Review:

...

GENERATED CONTENT

...

The SmartX Pro is a top-tier smartphone that offers a premium experience to users. Its stunning OLED display is a standout feature, delivering vibrant colors, deep blacks, and excellent contrast. The octa-core processor ensures smooth performance, even when running demanding apps and games. The battery life is impressive, lasting a full day on a single charge, and the fast-charging support is a convenient addition. However, the SmartX Pro is slightly expensive compared to other smartphones with similar specifications. Additionally, while the camera performs well in daylight conditions, it could be improved for low-light photography. Overall, the SmartX Pro is a solid choice for users seeking a high-end smartphone with a great display and strong performance.

...

In this example, the 'Detailed Review:' prefix serves as an output indicator, prompting the model to generate a comprehensive product review based on the provided instructions and context. The information preceding the output indicator, including the 'Pros,' 'Cons,' and 'Overall Rating,' acts as guidance for the model, highlighting the key aspects to focus on in the generated review. By leveraging this information, the model creates a detailed and coherent review that elaborates on the mentioned points, providing valuable insights for potential customers.

In the previous example, we used the Output Indicator Pattern to generate a detailed review. Now, let's explore how the Template Pattern can be used to generate content with a specific structure, such as a recipe.

3.4.3 Template Pattern (Aka. Fill-in-the-Blanks Pattern)

The Template Pattern, also known as the fill-in-the-blanks pattern, provides a structured template for the language model to complete. This pattern is useful when you have a specific structure in mind for the generated content. Unlike the 'Input Parameters' section, where the human fills in the blanks, in the Template Pattern, the AI fills in the blanks within the provided template structure. The template uses square brackets to indicate where the model should fill in the content.

3.4.4 Practical Example 2: Generating a Recipe

Let's say you want to generate a recipe using a template.

PROMPT

```
...  
  
Recipe: [Fill in the recipe name]  
Ingredients:  
- [Fill in ingredient 1]  
- [Fill in ingredient 2]  
- [Fill in ingredient 3]  
  
Instructions:  
1. [Fill in step 1]  
2. [Fill in step 2]  
3. [Fill in step 3]  
...
```

GENERATED CONTENT

...
Recipe: Chocolate Chip Cookies

Ingredients:

- 2 cups all-purpose flour
- 1 tsp baking soda
- 1 cup butter, softened
- 1 cup white sugar
- 1 cup brown sugar
- 2 large eggs
- 2 tsp vanilla extract
- 2 cups semisweet chocolate chips

Instructions:

1. Preheat the oven to 375°F (190°C). Line a baking sheet with parchment paper.
2. In a medium bowl, whisk together the flour and baking soda. Set aside.
3. In a large bowl, cream together the butter, white sugar, and brown sugar until light and fluffy. Beat in the eggs one at a time, then stir in the vanilla. Gradually mix in the flour mixture until just blended. Stir in the chocolate chips.
4. Drop rounded tablespoons of the cookie dough onto the prepared baking sheet, about 2 inches apart.
5. Bake for 9-11 minutes, or until the edges are lightly browned. Allow the cookies to cool on the baking sheet for 5 minutes before transferring them to a wire rack to cool completely.

...

The template pattern outlines the sections for a recipe, with placeholders for the name, ingredients list, and numbered instructions. The language model generates the complete recipe content to fill this template structure. It composes an original name, lists the required ingredients, and authors the step-by-step cooking instructions. The result is a coherent, formatted recipe output produced by the model to fit the provided template.

3.4.5 Output Style and Tone

Output Style and Tone refers to the specific writing style, persona, or voice you want the model to adopt when generating content. This can include emulating the style of a well-known figure, a celebrity, a particular genre, or even a fictional character. By specifying the desired style and tone, you can guide the model to generate content that matches a specific voice, personality, or tone, making it more engaging, authentic, and tailored to your target audience.

For example, if you're writing a blog post about sustainable fashion, you might want the language model to use persuasive language and a sense of urgency, as if written by a passionate environmental activist. Or, if you're creating a children's story, you might want the model to adopt a whimsical and imaginative tone reminiscent of classic fairy tales.

To specify the desired output style and tone, provide examples or references that capture the essence of the intended voice or explicitly describe the style and personality traits you want the model to emulate.

3.4.6 Practical Example 3: Generating a Speech in the Style of a Famous Person

Suppose you want the language model to generate a speech in the style of Martin Luther King Jr.

PROMPT

```
...  
  
Generate a speech about the importance of equality, in the style of Martin Luther King Jr.  
Speech:  
...
```

GENERATED CONTENT

...

My fellow Americans, today I stand before you to speak about the paramount importance of equality in our society. Just as Dr. Martin Luther King Jr. once said, "Injustice anywhere is a threat to justice everywhere." We cannot truly progress as a nation until we embrace the fundamental truth that all men and women, regardless of their race, religion, or background, are created equal.

It is our duty, as citizens of this great nation, to work tirelessly towards creating a society where every individual is judged not by the color of their skin, but by the content of their character. We must break down the barriers of prejudice and discrimination that have plagued our communities for far too long.

Let us remember the words of Dr. King: "I have a dream that one day this nation will rise up and live out the true meaning of its creed: 'We hold these truths to be self-evident, that all men are created equal.'" It is time for us to turn this dream into a reality.

So let us march forward, hand in hand, towards a future where equality reigns supreme. Let us build a world where our children can grow up knowing that they are valued, respected, and loved, regardless of their differences. Together, we can create a society that truly embodies the principles of justice, freedom, and equality for all.

Thank you, and may God bless America.

...

The language model generates a speech that emulates the style and tone of Martin Luther King Jr., focusing on the theme of equality. The generated speech includes powerful rhetorical devices, historical references, and a call to action, all characteristic of Dr. King's speeches.

Specifying the desired output format, whether through the Output Indicator Pattern, the Template Pattern, or the Output Style and Tone, is crucial for generating content that meets specific requirements. By using these patterns and specifying the desired style, you can guide the model to generate content in specific formats, structures, or voices. Practicing the use of these techniques will help you create prompts that generate well-formatted and stylistically appropriate content.

3.4.7 Hands-On Practice

Try creating prompts using the Output Indicator Pattern, the Template Pattern, and specifying the Output Style and Tone for the following scenarios:

1. Generate a product review for a smartphone using the Output Indicator Pattern
2. Create a job posting for a software engineer position using the Template Pattern.
3. Generate a poem about nature in the style of Robert Frost using the Output Style and Tone.
4. Write a news article about a recent political event using the Output Indicator Pattern.