

# Create structured prompts: XML, JSON, and frameworks.

Mastering XML tags, JSON schema, and the COSTAR framework is key to structuring statements that produce reliable and consistent AI output across all models.

Mastering XML tags, JSON schema, and the COSTAR framework is key to structuring statements that produce reliable and consistent AI output across all models.

The biggest improvement in statement quality comes from structure. Not better wording. Not longer statements, but structure.

A structured command tells the AI exactly where each piece of information should be: where the instructions are, where the context is, where the examples are, and what the output format should be. This eliminates ambiguity that causes inconsistent results.

## XML Tag: Platform

XML tags are the most flexible structuring tool. They work across all major AI models, and Claude is specifically trained with XML in its training data.

### Basic structure

Bạn là một nhà phân tích tài chính chuyên về các công ty SaaS. Phân tích dữ liệu hàng quý sau đây và đưa ra những hiểu biết. Quý 1: Doanh thu: 2,1 triệu đô la, Tỷ lệ khách hàng rời bỏ: 4,2%, NRR: 112% Quý 2: Doanh thu: 2,4 triệu đô la, Tỷ lệ khách hàng rời bỏ: 3,8%, NRR: 118% Quý 3: Doanh thu: 2,9 triệu đô la, Tỷ lệ khách hàng rời bỏ: 3,1%, NRR: 125% 1. Tóm tắt (2 câu) 2. Ba xu hướng chính (và phân tích thay đổi) 3. Một rủi ro cần lưu ý 4. Một đề xuất nên bắt đầu

Compare this to the unstructured version: *"You are a financial analyst. Consider Q1: Revenue \$2.1 million, customer churn rate 4.2%, NRR 112%. Q2: \$2.4 million, 3.8%, 118%. Q3: \$2.9 million, 3.1%, 125%. Tell me about the trends and what we should focus on."*

Both contain the same information. The more structured version will yield more consistent and predictable results.

## Common XML tags

Tag	Purpose	For example
	Defining the role/specialization of AI	Nhà khoa h?c d? li?u c?p cao
	What should we do?	Phân lo?i các yêu c?u h? tr? này
	Basic information	Chúng tôi ?ã ra m?t phiên b?n 2.0 vào tu?n tr??c
	Rules and limitations	T?i ?a 200 t?
	Few-shot examples	Input: X ? Output: Y
	Desired format	JSON v?i các tr??ng.
	Data to be processed	[??t d? li?u ng??i dùng ? ?ây]

? **Quick check** : You have a prompt with instructions, context, user input, and desired output formatting—all in a single paragraph. What's the risk?

**Answer** : Artificial intelligence (AI) can misinterpret what constitutes instructions, context, and input data. For example, if your context includes "always double-check calculations," the AI ??might interpret it as instructions when it's actually a description of company policy. XML tags prevent this: Content is read as background information, content is treated as commands to be followed.

## JSON schema for output control.

When you need structured output that your code can parse, specify the JSON schema:

Trích xu?t thông tin liên h? t? email sau. Tr? v? m?t ??i t??ng JSON v? i các tr??ng sau: { "name": "H? và tên ??y ?? c?a ng??i g?i", "email": "?? a ch? email", "company": "Tên công ty (ho?c null n?u không ???c ?? c? p)", "request": "Tóm t?t m?t câu v? nh?ng gì h? mu?n", "urgency": "th? p | trung bình | cao" } Chào, tôi là Marcus Chen t? DataFlow Inc. Chúng tôi c?n tr? giúp kh?n c?p v? vi?c tích h?p API. H? th? ng s?n xu?t c?a chúng tôi ?ang g?p s? c? và chúng tôi c?n ai ?ó tham gia cu?c g?i trong vòng m?t gi?. S? ?i?n tho?i tr?c ti?p c? a tôi là marcus@dataflow.io.

Expected output:

```
{ "name": "Marcus Chen", "email": "marcus@dataflow.io", "company": "DataFlow Inc
?n tr? giúp kh?n c?p v? tích h?p API do h? th?ng s?n xu?t g?p s? c?
", "urgency": "cao" }
```

## COSTAR Framework

COSTAR provides a complete prompt framework that works for all tasks:

Ingredient	Its meaning	For example
Context?	Context and situation	"We are a B2B SaaS company, launching a new feature next month."
Objective?	Things you need	"Draft a notification email for existing customers"
Style?	Communication methods	"Professional yet friendly, similar to Stripe's communication style."
Tone (tone)	Emotional quality	"Excited but not overstated - confident, not in a sales pitch."
Audience?	Who will receive it?	"The current customer is using the Pro package"
Response?	Formatting parameters	"Email subject line + 3-paragraph email body + Call-to-action (CTA) button text"

## COSTAR in practice

Công ty chúng tôi (TechCorp) bán phần mềm quản lý dự án. Chúng tôi vừa bắt đầu tính năng ưu tiên nhiệm vụ dựa trên AI. Người dùng thử nghiệm beta rất thích tính năng này (tỷ lệ hài lòng 92%). Chúng tôi sẽ ra mắt chính thức vào tháng Ba tới đây. Viết email thông báo sản phẩm mới tin và hào hứng, nhưng không quá ồn ào. Không dùng từ "mang tính cách mạng" hoặc "thay đổi cuộc chơi" - hãy tập trung nói lên giá trị của nó. Khách hàng hiện tại của gói Pro sẽ được sản phẩm trong vòng 3 tháng. Hãy nhắc nhở giao dịch. - Tiêu đề email (dưới 50 ký tự) - Nội dung vắn tắt trước tiêu đề (dưới 100 ký tự) - Nội dung email (3 dòng, mỗi dòng 2-3 câu) - Một nút CTA kèm với nội dung

**Quick check** : Two prompts generate the same content, but one uses COSTAR and the other only a paragraph. In which case is COSTAR MORE IMPORTANT?

**Answer** : When a prompt is reusable—used by a team, embedded in an application, or run hundreds of times—a disposable, personal prompt might not be formal. But a prompt running in a production environment or shared within a team needs to produce consistent quality regardless of who runs it. COSTAR's clear components prevent this discrepancy: Every team member uses the same context, style, and output specifications.

## Combining structures

The most powerful prompts combine XML tags with COSTAR principles and JSON output:

Bạn là một nhà phân tích phần mềm khách hàng. Bạn xem lý các đánh giá sản phẩm và trích xuất thông tin chi tiết có cấu trúc. Chúng tôi bán một ứng dụng quản lý dự án. Các đánh giá trên App Store. Phân tích đánh giá sau và trích xuất dữ liệu có cấu trúc. { "sentiment": "positive | negative | mixed", "topics": ["danh sách các tính năng yêu cầu", "feature\_requests": ["bắt đầu tính năng nào người dùng muốn"], "bugs\_mentioned": ["bắt đầu lỗi nào cần báo cáo"], "summary": "tóm tắt một câu" } {{REVIEW\_TEXT}}

The placeholder text { {REVIEW\_TEXT} } turns this into a template—you can replace different reviews while the structure remains the same. This is how prompts in production work: Define the structure once, only change

the input.

## Practice exercises

1. Take a prompt you frequently use and restructure it using XML tags.
2. Apply the COSTAR framework to a complex task (e.g., writing a proposal, data analysis).
3. Create a JSON output schema to extract information from unstructured text.
4. Compare the output of the old question with the structured version – note any differences in consistency.

## Key points to remember

1. Structure is more important than length - well-organized prompts produce more consistent results than long, unstructured prompts.
2. XML tags create a clear distinction between instructions, context, examples, and output formatting.
3. JSON schema ensures that the output can be parsed by code—essential for production applications.
4. COSTAR (Context, Objective, Style, Tone, Audience, Response) includes all aspects of a complete prompt.
5. Templates with placeholders ({{ VARIABLE }}) allow structured prompts to be reused across different inputs.
6. Claude was trained with XML tags; all main models respond well to structured prompts.

### 1. Question 1:

In the COSTAR framework, what does the 'S' (Style) component do?

1. A. It sets the font style.
2. B. It tells the AI the generation rate - confusing correlation and cause here leads to ineffective strategies.
3. C. It determines HOW AI should communicate.

EXPLAIN:

Style relates to the method of communication, not the content. The same information presented in a "leadership summary" style versus a "detailed technical document" will produce significantly different results—even if the data is identical. Style dictates the AI's tone: word complexity, sentence length, use of technical jargon, level of detail, and rhetorical approach. Formal or informal, technical or easy to understand, persuasive or neutral—all shape word choices, sentence structure, and overall tone.

### 2. Question 2:

When should you use JSON formatting instead of XML in your prompt?

1. A. Always - JSON is always better - although this is partly true, it oversimplifies the actual picture.
2. B. When you need AI output that can be directly coded and parsed (API, database, application)
3. C. Never - XML is always better.

EXPLAIN:

JSON is the right choice when AI output is fed into another system. If a Python script is processing AI responses, the JSON schema ensures consistent, parsable output. XML is better for human-readable prompts, where you're organizing instructions. Many developers use XML for prompt structure and JSON for output formatting—combining both strengths. JSON ensures output conforms to a defined schema that programs can use without additional parsing.

3. Question 3:

Why do XML tags improve prompt reliability?

1. A. They create clear boundaries between different parts of the prompt.
2. B. AI models that can only read XML.
3. C. They make the prompt look more professional - this sounds reasonable but overlooks a crucial element that makes all the difference.

EXPLAIN:

XML tags solve the boundary problem. Without them, a long prompt would be a jumbled block of text where instructions, context, and examples are all mixed together. With tags like `<<`, `<>`, and `>>`, each part is clearly delineated. The AI knows exactly what is instruction, what context to infer, and what example to follow. Instructions, context, examples, and output formatting are clearly separated so the AI cannot confuse one with the other.

Submit your work

## Training results

You have completed **0** questions.

-- / --

Review the lesson

You finished reading the article "**Create structured prompts: XML, JSON, and frameworks.**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.