Multi-round Conversation

This guide will introduce how to use the DeepSeek `/chat/completions` API for multi-turn conversations.

The DeepSeek `/chat/completions` API is a "stateless" API, meaning the server does not record the context of the user's requests. Therefore, the user must **concatenate all previous conversation history** and pass it to the chat API with each request.

The following code in Python demonstrates how to concatenate context to achieve multi-turn conversations.

```python
from openai import OpenAI
client = OpenAI(api_key="<DeepSeek API Key>", base_url="https://api.deepseek.com")

# Round 1
messages = [{"role": "user", "content": "What's the highest mountain in the world?"}]
response = client.chat.completions.create(
    model="deepseek-chat",
    messages=messages
)

messages.append(response.choices[0].message)
print(f"Messages Round 1: {messages}")

# Round 2
messages.append({"role": "user", "content": "What is the second?"})
response = client.chat.completions.create(
    model="deepseek-chat",
    messages=messages
)

messages.append(response.choices[0].message)
print(f"Messages Round 2: {messages}")
```

In the **first round** of the request, the `messages` passed to the API are:

```
[
    {"role": "user", "content": "What's the highest mountain in the world?"}
]
```

In the **second round** of the request:

1. Add the model's output from the first round to the end of the `messages`.
2. Add the new question to the end of the `messages`.

The `messages` ultimately passed to the API are:

```
[
    {"role": "user", "content": "What's the highest mountain in the world?"},
    {"role": "assistant", "content": "The highest mountain in the world is Mount Everest."},
```

```
    {"role": "user", "content": "What is the second?"}
]
```