

What are generics in Python? Why we use it for TContext?

Generics are like **empty boxes** that can hold **any type of thing** — but you don't have to say what kind of thing right away.

Imagine you want to write a function or class that works with **different types** (like numbers, text, or custom objects) — instead of writing separate code for each type, you use **generics** to make your code flexible.



What is TContext and Why Do We Use It?

In the **OpenAI Agents SDK**, TContext is just a **name for “any context type.”**

- Every agent might have **different data** it needs to remember or work with.

For example:

- A shopping agent may need a CartContext
- A chat agent may need a ChatContext

So instead of writing separate agents for each one, we use a **generic TContext**.



This helps with:

- **Flexibility** — same agent code works for different tasks.
- **Safety** — Python can check you're using the right type of context.
- **Clean Code** — no need to repeat similar code for every type.

 **Think of it like this:**

If an agent is a **chef**, and the context is the **recipe**, then:

TContext means: "Hey chef, you'll get *some* recipe, but I'll tell you what type of recipe it is later!"