COMP1531

5.4 - SDLC Development - Persistence

Data

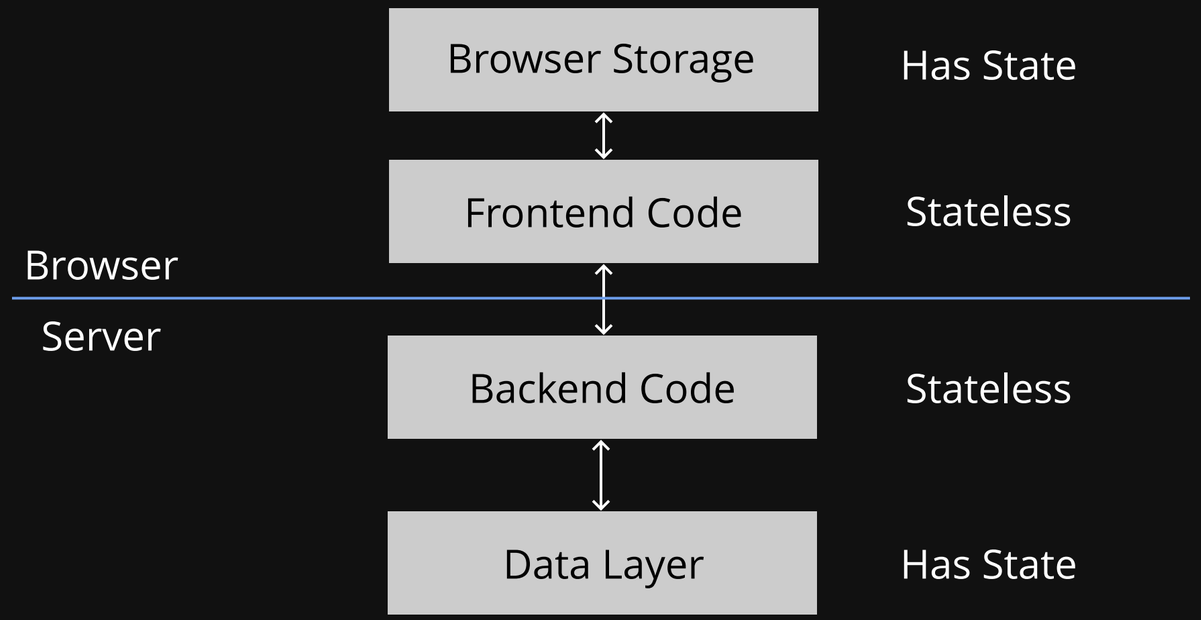
Data: Facts that can be recorded and have implicit meaning

Data is one of the fastest and most rapidly growing areas within software.

From **data (raw)** was can find **insights (information)** that allow us to make **decisions**.

Data Layer

Data Layer: The part of the tech stack that provides persistence



Databases

Data is only as powerful as you can store and access it. Study COMP3311 to learn more about efficient data storage.

There are 3 main ways to store data:

1. In-memory (non-persistent)
2. In-file
3. In-database (SQL)

As you move down the list, barrier to entry becomes higher, but so does performance.

In COMP1531 we will only explore (2)

Storing Data: Persistence

Persistence: When program state outlives the process that created it. This is achieved by storing the state as data in computer data storage

What is storage?

- CPU cache?
- RAM?
- Hard disk? (we usually mean this one)

Storing Data: Persistence

Most modern backend/server
applications are just source code + data

Storing Data: In practice

A very common and popular method of storing data in python is to "pickle" the file.

- Pickling a file is a lot like creating a .zip file for a variable.
- This "variable" often consists of many nested data structures (a lot like your iteration 2 data)

Storing Data: In practice

Let's look at an example

pickle_it.py

unpickle_it.py