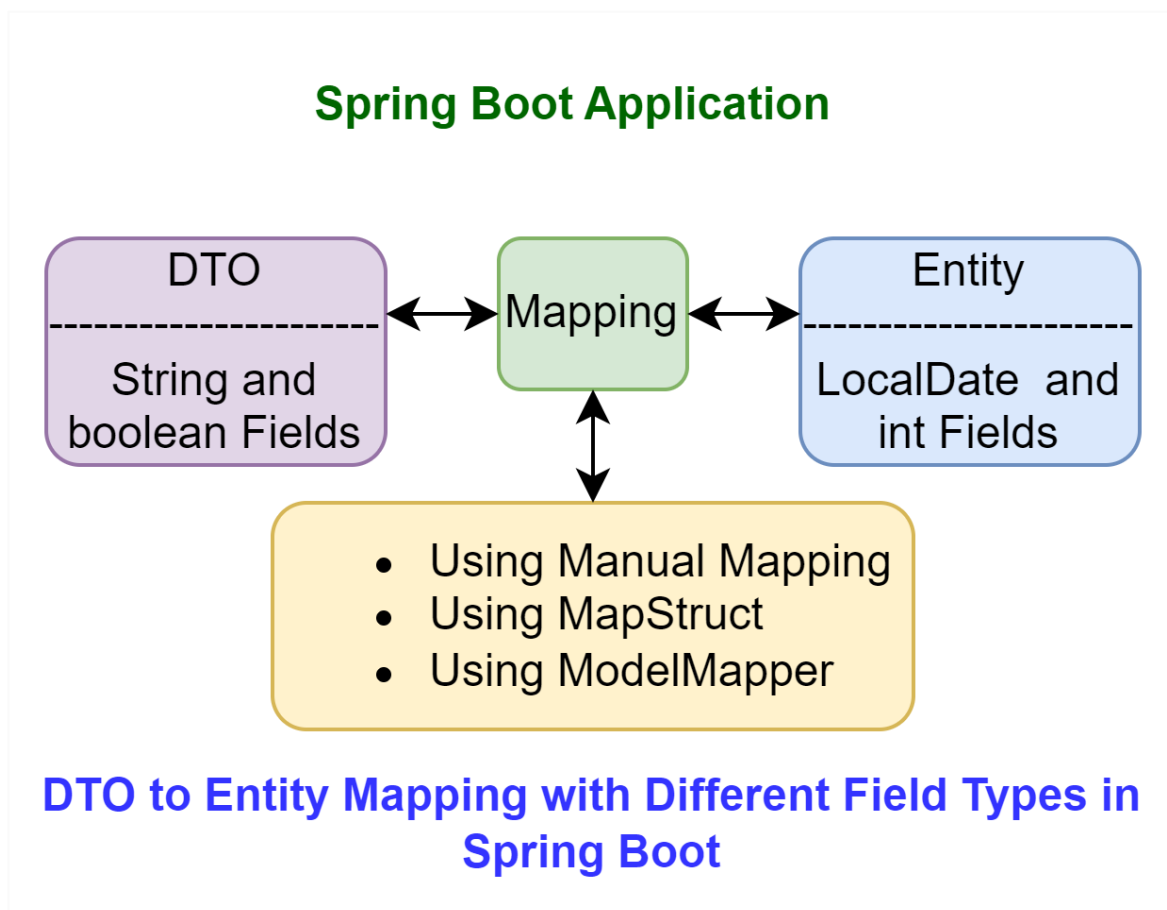


What is a DTO (Data Transfer Object)?

A **DTO** is a simple object used to **transfer data between different layers of an application** (usually between the controller and service layers, or from service to client).

Why use a DTO?

- **Security:** You may not want to expose the entire database entity to the client (e.g., passwords, IDs).
- **Efficiency:** You only send the needed data, which makes it faster.
- **Separation of concerns:** Keeps the internal model (Entity) separate from what is exposed outside.



Imagine you are **developing** an application for an **online store**. You have an **entity** called **Product**, which **includes** a lot of information: **name**, **description**, **price**, **production date**, **discounts**, and so on.

A **client** requesting a **list** of products **doesn't need all** of this information. Instead, you can create a **DTO object** that contains only the **necessary fields** (**such as the name and price**) to **send** this **data** to the **client**.

The **map** **method** from the **ModelMapper** **library** automatically **maps the fields** of one **object** to the **fields** of another based on their **names** and **types**.