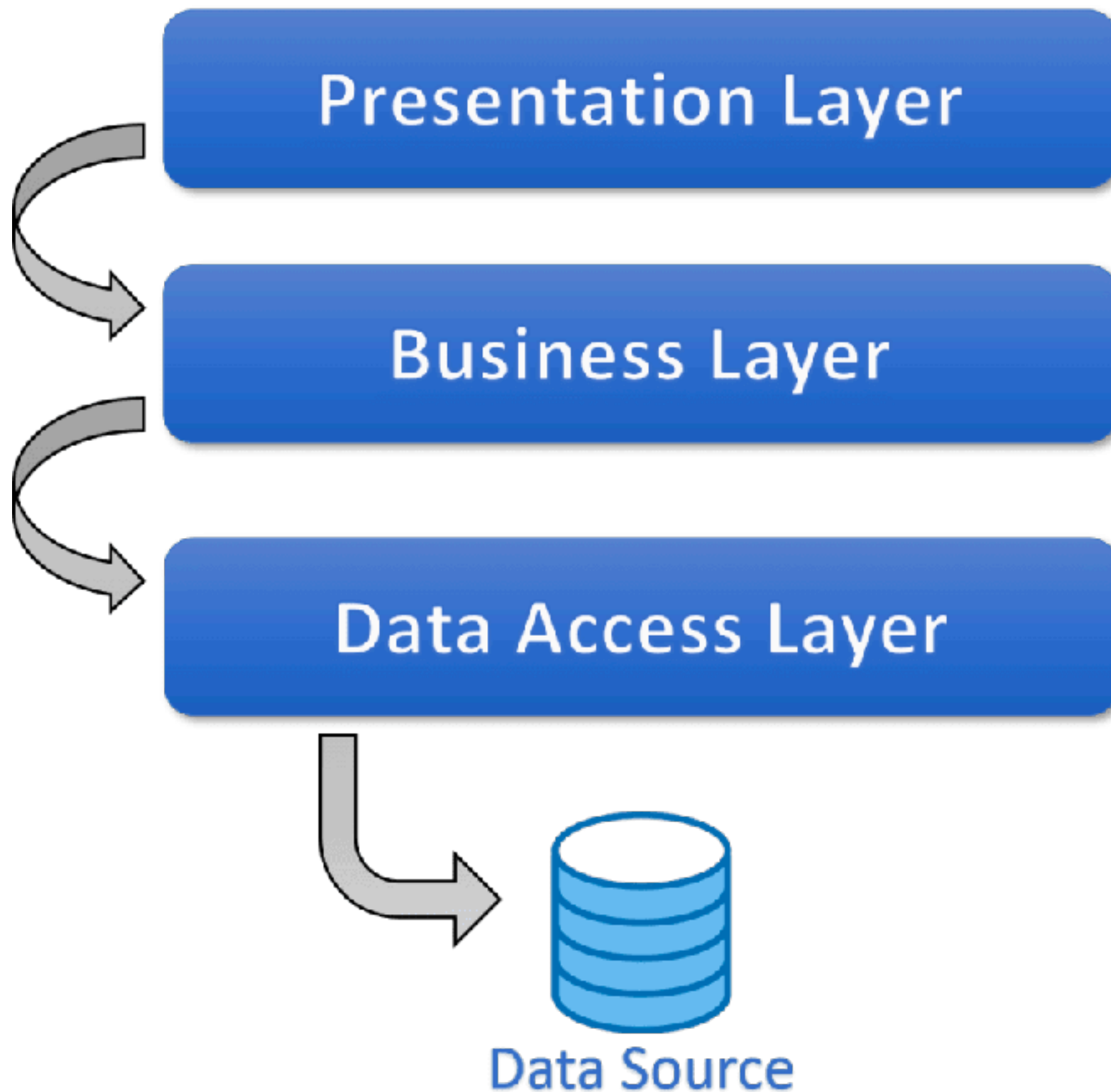
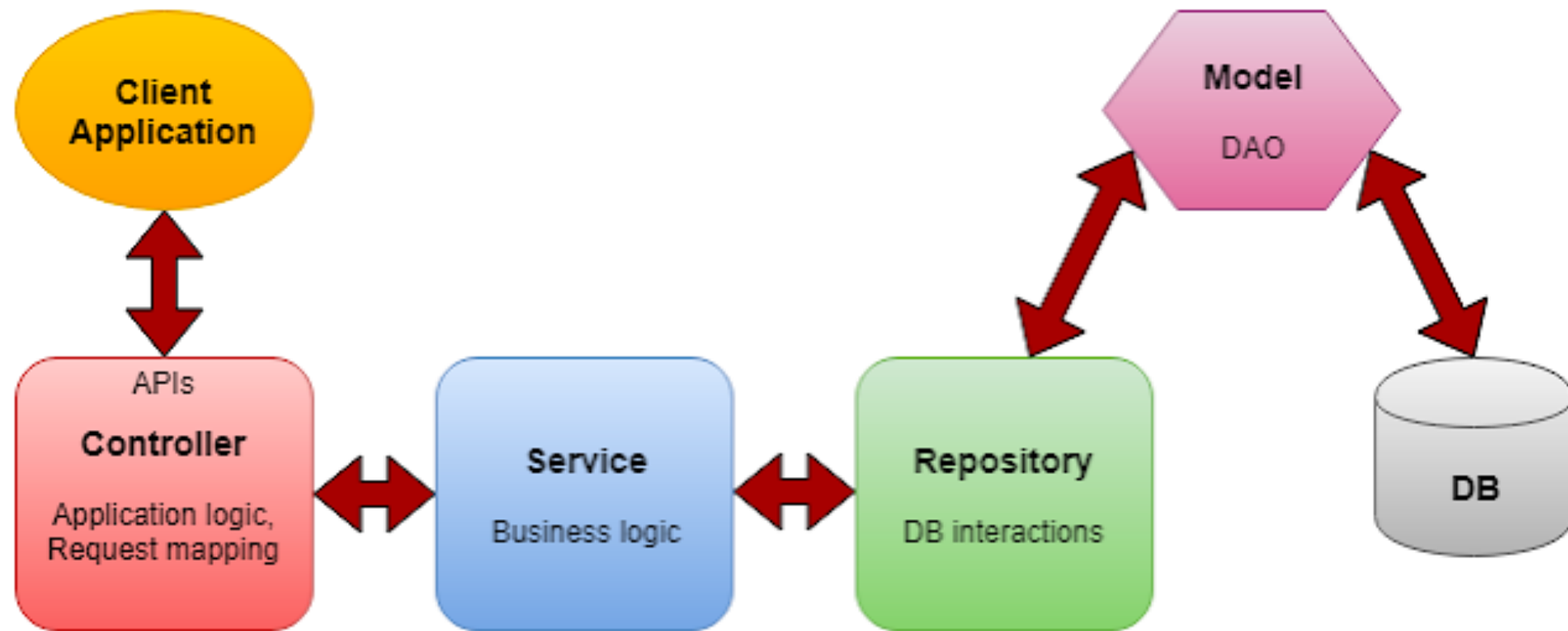


# Architecture Pattern

Layered Architecture Pattern





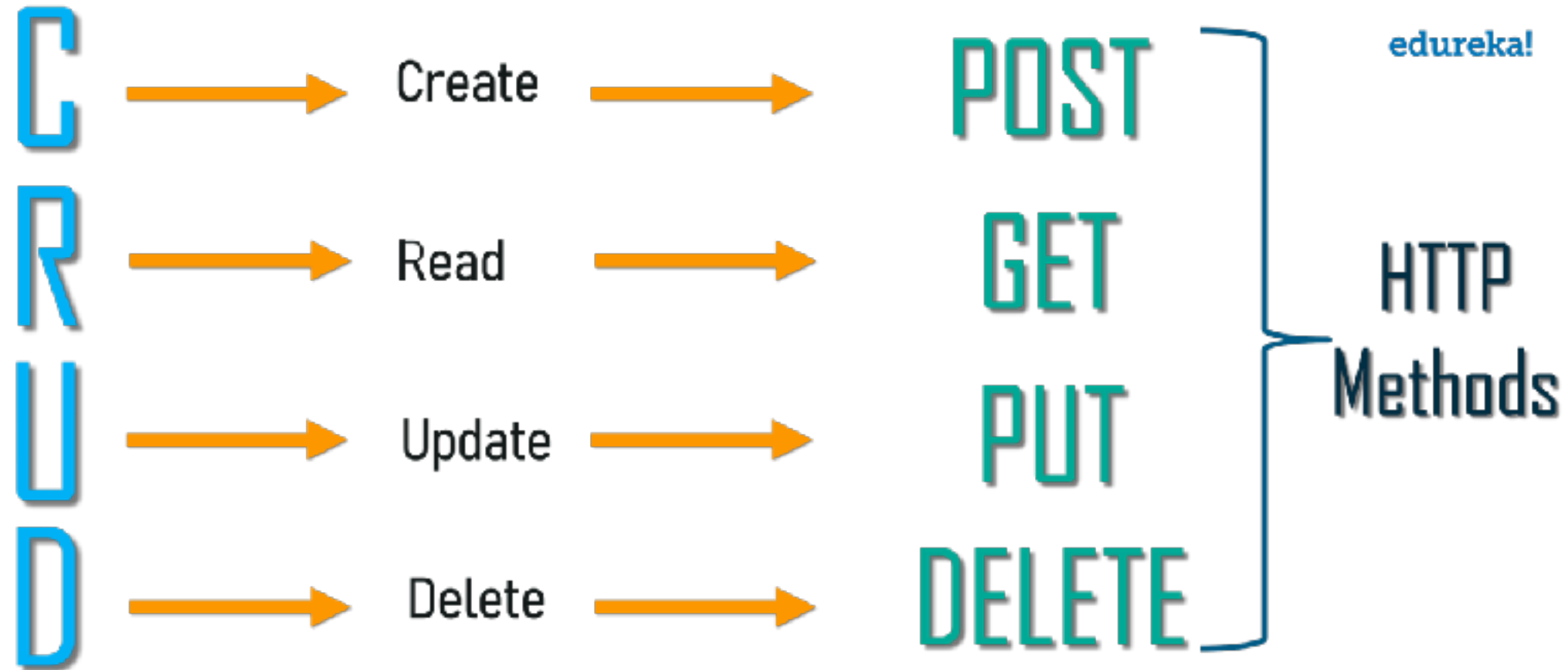
# Setup

1. Spring starter: <https://start.spring.io/>
2. Download MySQL community server: <https://dev.mysql.com/downloads/mysql/>
3. Download MySQL workbench community edition: <https://dev.mysql.com/downloads/workbench/>

# Database setup

- **Create a database:** *create database trial\_db;*
- **Use database:** *use trial\_db;*
- **Create table:** *create table Account (  
    Number int,  
    Name varchar(255),  
    Balance long,  
    primary key (Number)  
);*
- **Add Values to table:** *insert into Account(Number, Name, Balance)  
    values  
    (100, "Maya", 10000),  
    (101, "Riya", 12000),  
    (102, "Arya", 15000);*
- **Select statement:** *select \* from Account;*

# CRUD Operations



# SpringBoot Basics

- Spring Boot is an open source java based framework.
- Main Concepts:
  1. Dependency Injection
  2. Inversion Of Control
  3. Aspect Oriented Programming

# Spring boot annotations

- @SpringBootApplication - to annotate the main class
- @RestController - to denote web controllers that returns JSON or XML data
- @Service - used to denote classes that store business logics
- @Entity - help us to map our domain objects (POJOs) to the relational database tables.
- @Id - to denote the unique identifier
- @Autowired - used for automatic dependency injection
- @Repository - used to denote repositories



# Spring Boot annotations - II

- @RequestMapping
- @RequestParam
- @GetMapping
- @PostMapping
- @DeleteMapping
- @Query
- @Modifying
- @Transactional