

Bài 5

Thao tác với CSDL và ORM

Module: BOOTCAMP WEB-BACKEND DEVELOPMENT

Kiểm tra bài trước

Hỏi và trả lời về các khó khăn gặp phải trong bài "Data Binding Form" Tóm tắt lại các phần đã học từ bài "Data Binding Form"

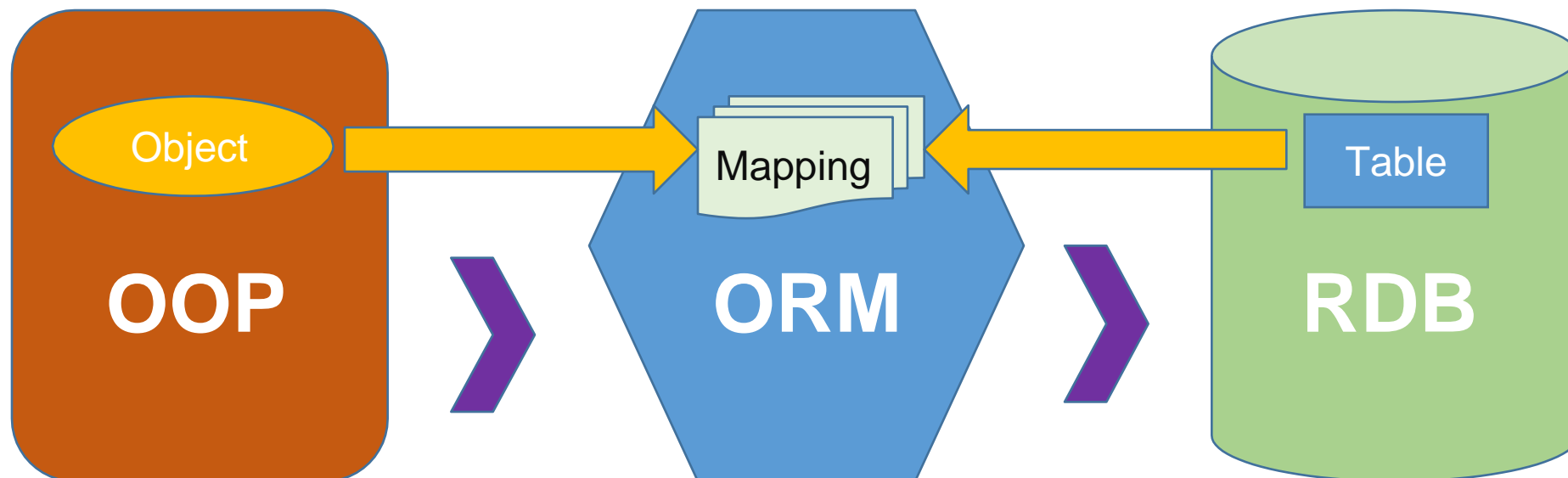
- Trình bày được khái niệm ORM

Thảo luận

ORM

ORM

- ORM (Object-Relational Mapping) là kỹ thuật liên kết giữa các đối tượng trong lập trình với các đối tượng trong CSDL
- ORM cho phép truy xuất dễ dàng đến dữ liệu thông qua các đối tượng lập trình
- ORM giúp lập trình viên tập trung thao tác với các đối tượng, không cần quá quan tâm đến CSDL thực tế đang dùng



Các góc nhìn khác nhau về dữ liệu

- Góc nhìn từ quản trị dữ liệu

```
CREATE TABLE persons (  
    id integer NOT NULL,  
    name varchar(50) NOT NULL,  
    salary float, PRIMARY KEY(id)  
);  
INSERT INTO persons (id, name) VALUES  
(1, 'John Doe');  
UPDATE persons SET salary=2000 WHERE  
id=1;
```

- Góc nhìn từ lập trình viên

```
public class  
    Person {  
        public String  
        name; public  
        float salary;  
        public Person(String name) { ... }  
    }  
    Person p = new  
    Person("John Doe");  
    PersistenceLayer.save(p);  
    p.setSalary(2000);  
    PersistenceLayer.update(p);
```

Ưu điểm và nhược điểm của ORM

- Ưu điểm:

- Quản lý dữ liệu tập trung trong code
- Các thao tác với dữ liệu được thực hiện tự động
- Tránh được các lỗi cú pháp SQL
- Hỗ trợ giao dịch (transaction)
- Có thể cache dữ liệu để truy xuất nhanh hơn

- Nhược điểm:

- Đối với các dự án lớn, hiệu năng truy xuất dữ liệu thường hạn chế
- Lập trình viên dễ bị rơi vào bẫy truy xuất dữ liệu quá nhiều (vì quá dễ để truy xuất), ảnh hưởng đến hiệu năng của hệ thống
- Đối với các thao tác phức tạp, có thể cần đến việc sử dụng SQL thuần

Một số ORM framework dành cho Java

- Apache Cayenne
- ActiveJDBC
- EclipseLink
- Ebean
- OpenJPA
- Hibernate
- JPA (Java Persistence API)

Tổng kết

- ORM là cơ chế ánh xạ dữ liệu giữa môi trường thực thi và CSDL

Hướng dẫn

Hướng dẫn làm bài thực hành và bài tập

Chuẩn bị bài tiếp theo: JPA