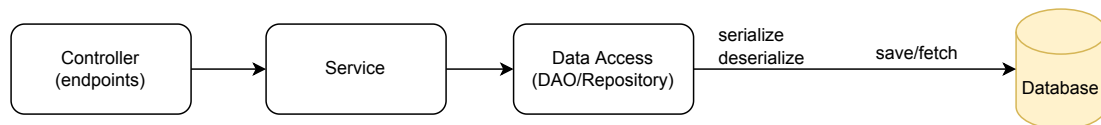
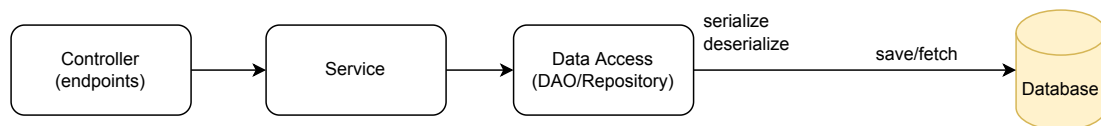
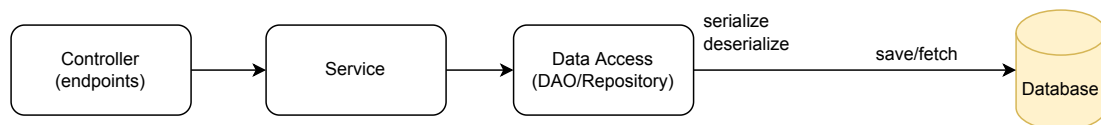
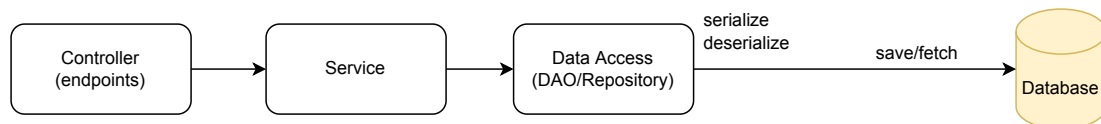
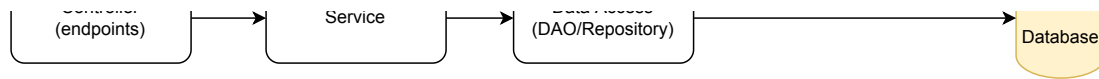


- create connection
- load driver
- exchange data
- update/commit
- update/rollback
- close connection

JDBC**ORM: Object Relational Mapping****Hibernate (JDBC)****Spring JDBCTemplate (JDBC)****Spring + Hibernate (JDBC)****JPA: Java Persistence API****Spring + Hibernate (JDBC)**

2/18/22, 5:54 PM

Day-7_JPA



Spring Data

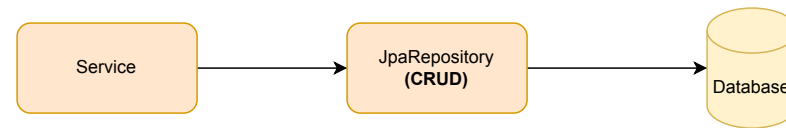
Spring Data JPA

Spring Data Mongo

Spring Data Rest

Spring Data JPA

JpaRepository -> PagingRepository -> CrudRepository



Embeddable

```

public class Person {

    @Id
    private int id;

    private String name;

    private Address address;

    @AttributeOverride("billing_street", "billing_zipcode", "billing_city")
    private Address billingAddress;

}

```

person_table:
id, name, street, zipcode, city, billing_street, billing_zipcode, billing_city

```

public class Address {
    private String street;
    private String zipcode;
    private String city;
}

```

Secondary Table

```

public class Person {
    private int id;
    private String name;

    // address
    private String city;
    private String zipCode;
    private String street;

    // phone
    private String areaCode;
    private String phoneNumber;

    private List<Ticket> ticketList;

    private Map<String, String> attributes;
}

```

```

public class Ticket {
    private String number;
}

```

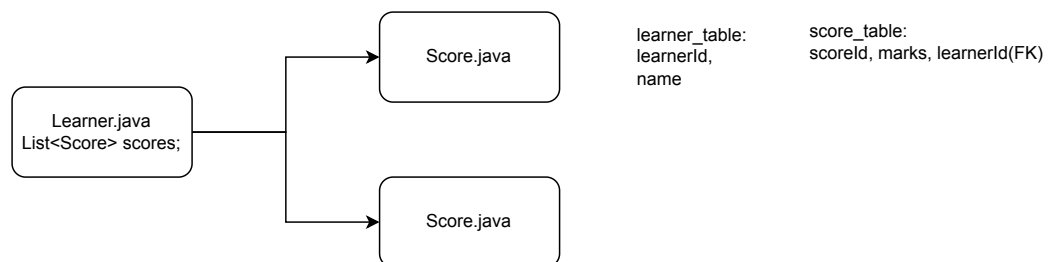
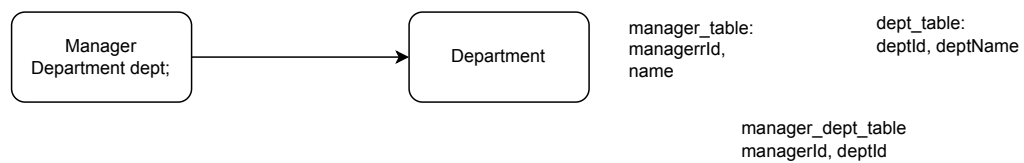
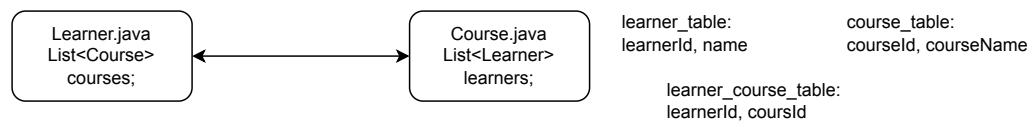
person_table:
id, name

address_table:
city, zipcode, street, person_id

phone_table:
areaCode, phoneNumber, person_id

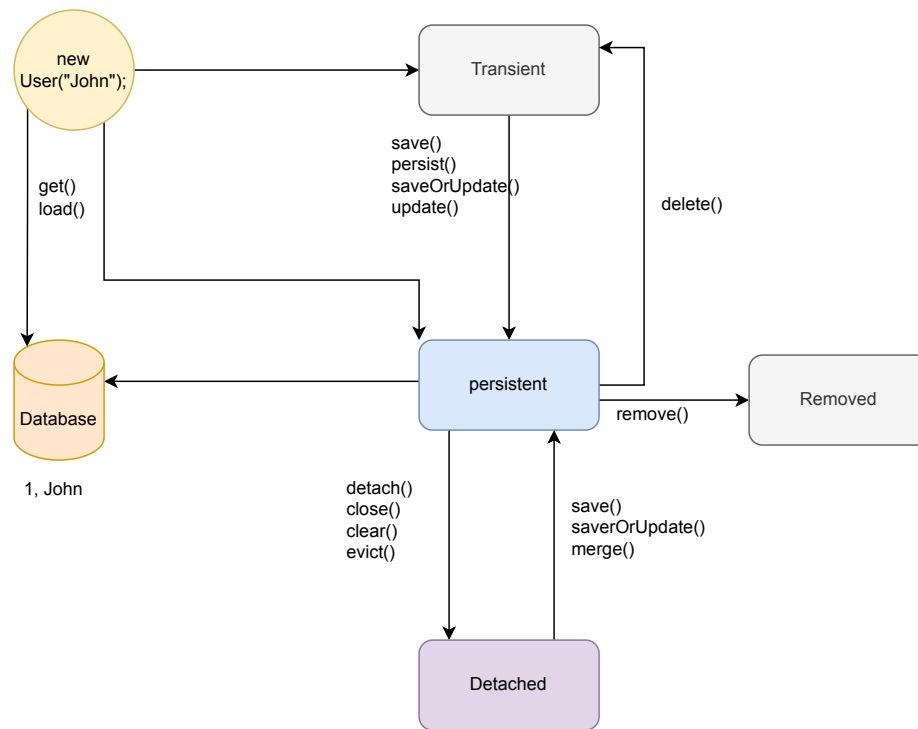
ticket_table:
number, person_id

attributes_table:
attr_name, attr_value, person_id

Associations**One-to-Many / Many-to-One****One-to-One****Many-to-Many**

States in Hibernate

```
/**
 * States of Objects in Hibernate/JPA:
 * 1) Transient
 * 2) Detached
 * 3) Persisted
 * 4) Remove
 *
 */
```



Inheritance in Hibernate/JPA

```
class Animal {  
    int id;  
}
```

```
class Cat extends Animal {  
    String meow;  
}
```

```
class Dog extends Animal {  
    String bark;  
}
```

1) Single Table

id, meow, bark, identifier
1, meow, null, cat
2, null, bark, dog

2) Table Per Class (Join)

animal table:	cat table:	dog table:
id	id, meow	id, bark
1	1, meow	2, bark
2		

3) Table Per Concrete Class

cat table:	dog table:
id, meow	id, bark
1, meow	1, bark

