

Skillbox

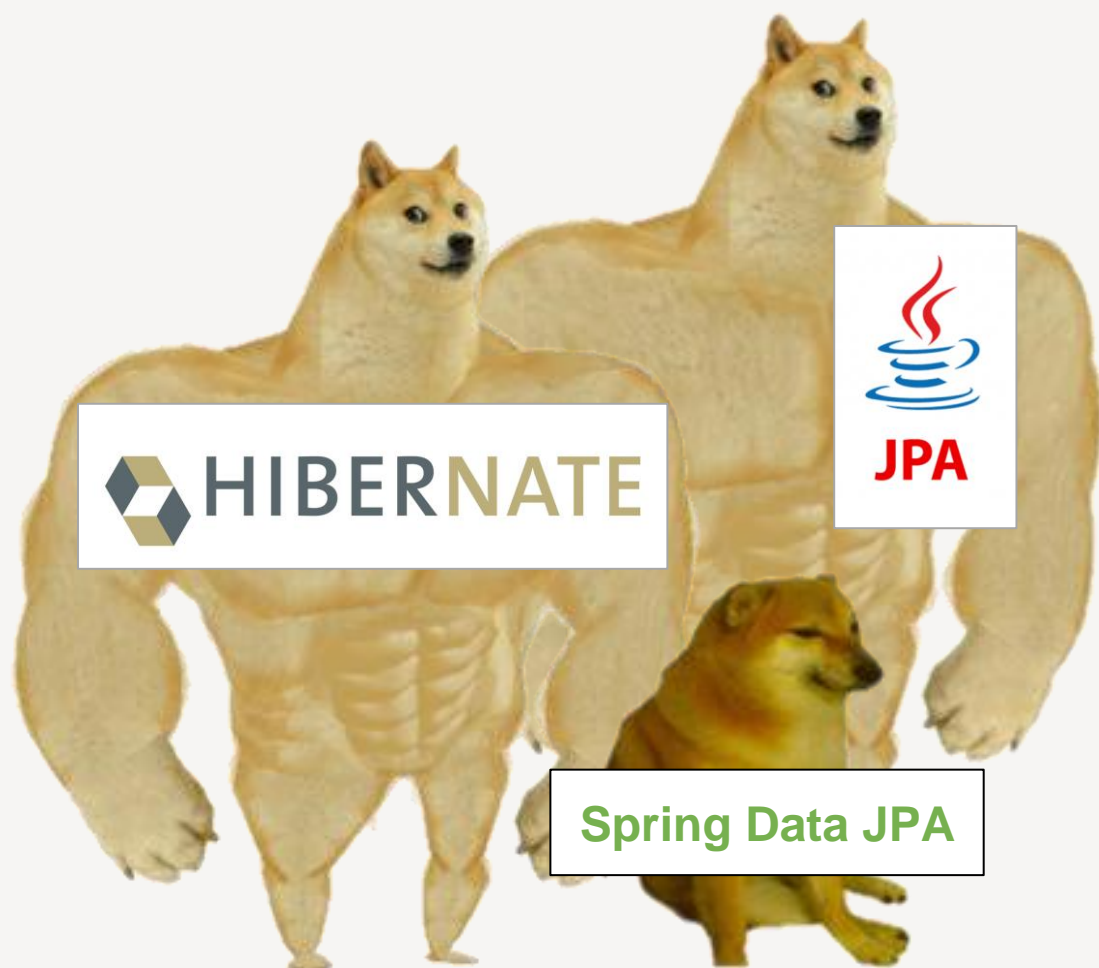
Spring Data JDBC

Кирилл Кошаев

Java-разработчик в «Газпром информ»

Spring Data JDBC

Spring Data JDBC — это фреймворк, фокусирующийся на более простой модели ORM, то есть не на такой сложной и менее навороченной, чем JPA/Hibernate



Я тоже ORM
фреймворк!



spring-boot-starter

Для Spring Data JDBC в Spring Boot доступна специальная стартер-зависимость spring-boot-starter-data-jdbc

Spring Data JDBC не поддерживает генерацию схемы БД

```
<dependency>  
  <groupId>org.springframework.boot</groupId>  
  <artifactId>spring-boot-starter-data-jdbc</artifactId>  
</dependency>
```



schema.sql

```
DROP TABLE IF EXISTS books;
```

```
CREATE TABLE books(  
  id INT AUTO_INCREMENT PRIMARY KEY,  
  author VARCHAR(250) NOT NULL,  
  title VARCHAR(250) NOT NULL,  
  price_old VARCHAR(250) DEFAULT NULL,  
  price VARCHAR(250) DEFAULT NULL  
);
```

Spring Data JDBC ENTITY

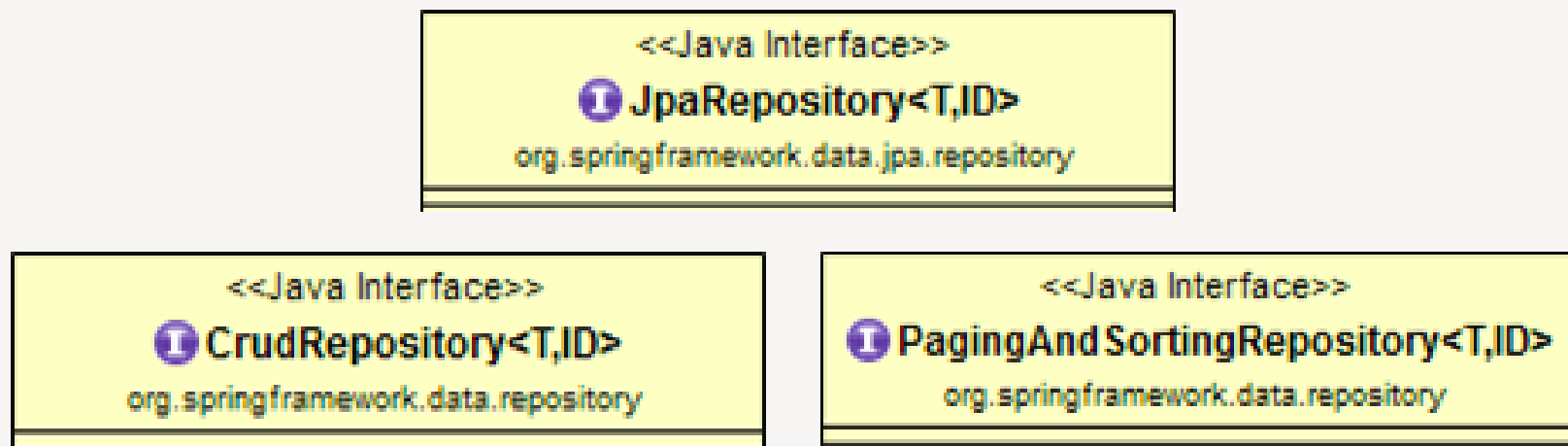
Как и в других проектах Spring Data, мы используем аннотации для сопоставления POJO классов с таблицами базы данных. В Spring Data JDBC объект должен иметь @Id, именно её он использует для идентифицирования сущности

```
OrdinaryUser -> camelcase  
ordinary_user -> snake case
```

```
public class User {  
    @Id  
    Long id;  
    @Column("name")  
    String name;  
    String role;  
    //constructors getters setters  
}
```

«Repository»

Шаблон «репозиторий» — один из самых популярных шаблонов, связанных с темой persistence. Он инкапсулирует конкретные детали реализации хранилища данных и позволяет создавать бизнес-код на ещё более высоком уровне абстракции



```
@Repository
public interface UserRepository extends CrudRepository<User,
Long> {
    //custom query methods
}
```

Query methods

- Spring Data JDBC не поддерживает генерацию запросов на основе имени метода.
- Spring Data JDBC не работает с JPQL/HQL.
- Spring Data JDBC не поддерживает ссылки на параметры с номерами индексов. В текущей версии мы можем ссылаться на параметры только по имени.

`@Repository`

```
public interface PersonRepository extends CrudRepository<Person,  
Long> {
```

```
@Query("select * from person where first_name=:name")  
List<Person> findByFirstName(@Param("name") String name);
```

`@Modifying`

```
@Query("UPDATE person SET name = :name WHERE id = :id")  
boolean updateByName(@Param("id") Long id, @Param("name") String  
name);  
}
```

**Спасибо
за внимание!**