

1. Java dünyasındaki framework'ler ve çözdükleri problemler nedir? Kod Örneklendirini de içermelidir.

Some of the most popular Java frameworks include:

Spring Framework

Problem: Some programmers encounter challenges when creating apps. Handling databases, web apps, and security management can get complicated. Issues may arise during the development process.

Solution: Spring has useful features to address these problems. It includes Dependency Injection and Aspect Oriented Programming (AOP). These allow coders to write cleaner, more modular, and easily testable code.

Code Example:

@Controller

```
public class MyController {
```

```
    @Autowired
```

```
    private MyService myService;
```

```
    @RequestMapping("/hello")
```

```
    public String sayHello(Model model) {
```

```
        model.addAttribute("message", myService.sayHello());
```

```
        return "hello";
```

```
    }
```

```
}
```

Hibernate

Problem: Developers often struggle with managing the mismatch between Java objects and database tables.

Solution: Hibernate is an Object Relational Mapping (ORM) tool that solves this incompatibility issue. It maps Java objects to database tables, simplifying CRUD operations on data.

Code Example:

```
@Entity
@Table(name = "Users")
public class User {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;
    private String name;
    // getters and setters
}
```

Spring Boot

Problem: Setting up and starting Spring apps can sometimes be tricky. The configuration and initialization processes can introduce complexity.

Solution: Spring Boot simplifies these steps. It provides auto configuration and starter dependencies. This allows for quicker and easier setup and launching of Spring applications.

Code Example:

```
@SpringBootApplication
public class MyApplication {
    public static void main(String[] args) {
        SpringApplication.run(MyApplication.class, args);
    }
}
```