Burcu Kündüm

Q1 Java Frameworks and Their Solutions

A Java Framework is a structured collection of pre-existing code or code blocks utilized for developing applications using Java. Thanks to these reusable structures, developers can avoid writing code from scratch. These pre-prepared structures can be employed for establishing connections to software, hardware, or databases.

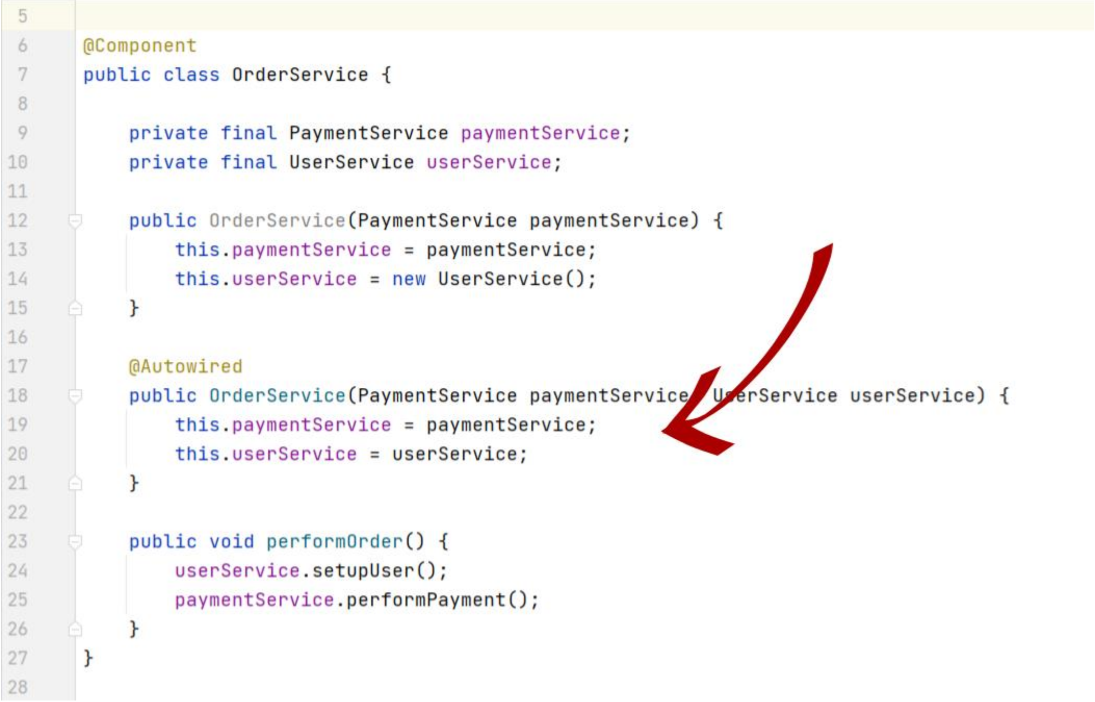
Various frameworks offer distinct advantages. Below, I outline some of the most popular frameworks along with the solutions they provide:

1) Java Spring Framework

It offers a wide range of solutions; commonly used examples are listed below.

1.1 Dependency Injection

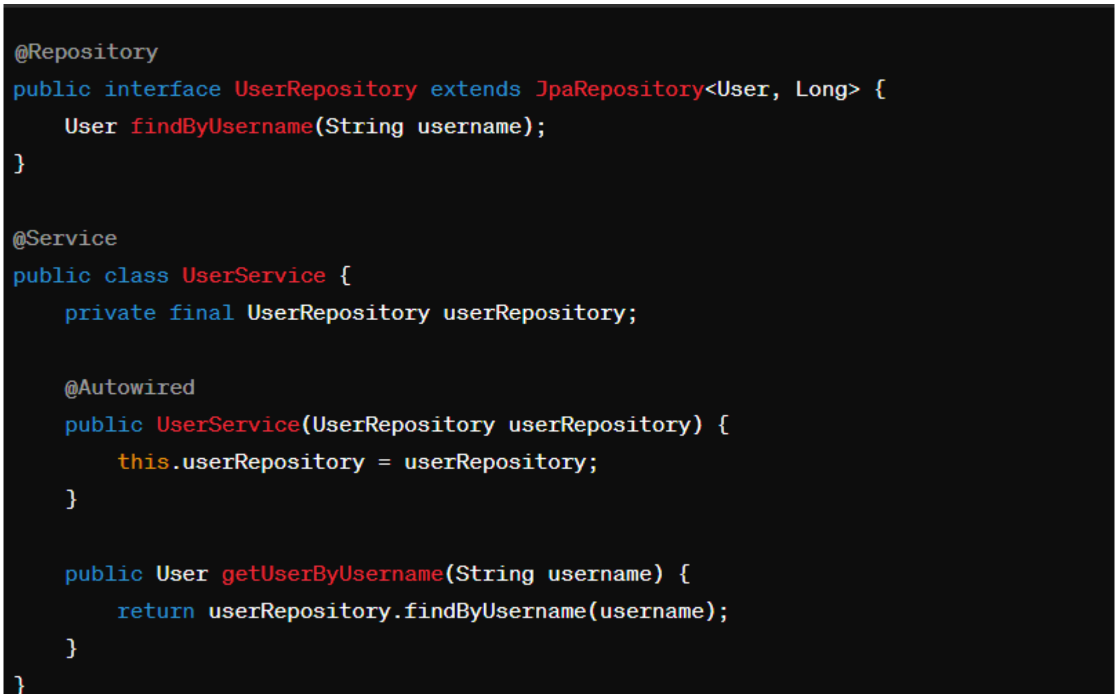
It reduces dependencies and makes code more flexible, reusable, and testable.



```java

@Component
public class OrderService {

    private final PaymentService paymentService;
    private final UserService userService;

    public OrderService(PaymentService paymentService) {
        this.paymentService = paymentService;
        this.userService = new UserService();
    }

    @Autowired
    public OrderService(PaymentService paymentService, UserService userService) {
        this.paymentService = paymentService;
        this.userService = userService;
    }

    public void performOrder() {
        userService.setupUser();
        paymentService.performPayment();
    }
}

```

1.2 Database Connection

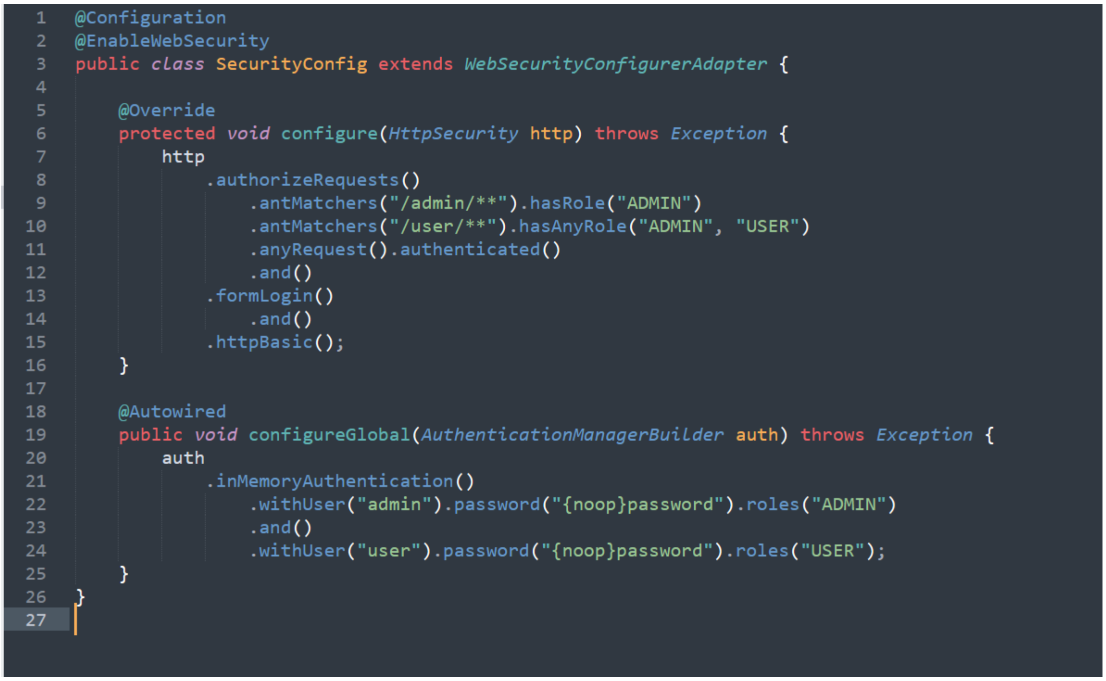
Ease of accessing the database using DBC or JPA.



```
@Repository
public interface UserRepository extends JpaRepository<User, Long> {
    User findByUsername(String username);
}

@Service
public class UserService {
    private final UserRepository userRepository;

    @Autowired
    public UserService(UserRepository userRepository) {
        this.userRepository = userRepository;
    }

    public User getUserByUsername(String username) {
        return userRepository.findByUsername(username);
    }
}
```

1.2 Security

Authentication, authorization and access control.



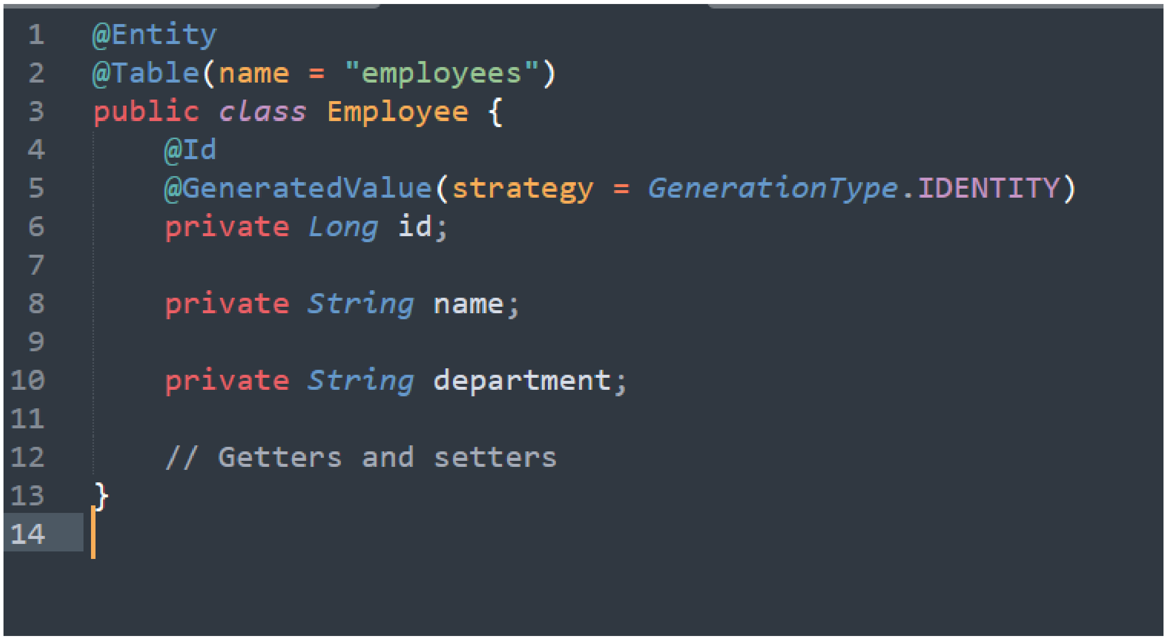
```
@Configuration
@EnableWebSecurity
public class SecurityConfig extends WebSecurityConfigurerAdapter {

    @Override
    protected void configure(HttpSecurity http) throws Exception {
        http
            .authorizeRequests()
                .antMatchers("/admin/**").hasRole("ADMIN")
                .antMatchers("/user/**").hasAnyRole("ADMIN", "USER")
                .anyRequest().authenticated()
                .and()
            .formLogin()
                .and()
            .httpBasic();
    }

    @Autowired
    public void configureGlobal(AuthenticationManagerBuilder auth) throws Exception {
        auth
            .inMemoryAuthentication()
                .withUser("admin").password("{noop}password").roles("ADMIN")
                .and()
                .withUser("user").password("{noop}password").roles("USER");
    }
}
```

2) Java Hibernate

It is an ORM (Object-Relational Mapping) framework used to manage database operations. Hibernate allows mapping Java objects to relational database tables and managing database operations.

2.1 Object-Relational Mapping - ORM)

It allows mapping database tables to Java objects.



```java
@Entity
@Table(name = "employees")
public class Employee {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long id;

    private String name;

    private String department;

    // Getters and setters
}
```

2.2 CRUD Operations

Accessing the database, adding, reading, updating and deleting data can be done.

2.3 HQL - Hibernate Query Language) and Criteria API

You can create custom queries to pull data from the database.

3) Apache Struts

Apache Struts is a web application framework for developing Java-based web applications. Struts makes web applications more modular and manageable using MVC (Model-View-Controller) architecture.

3.1 Model-View-Controller (MVC) Design Pattern:

It enables development of web applications by separating them into model, view and controller components.

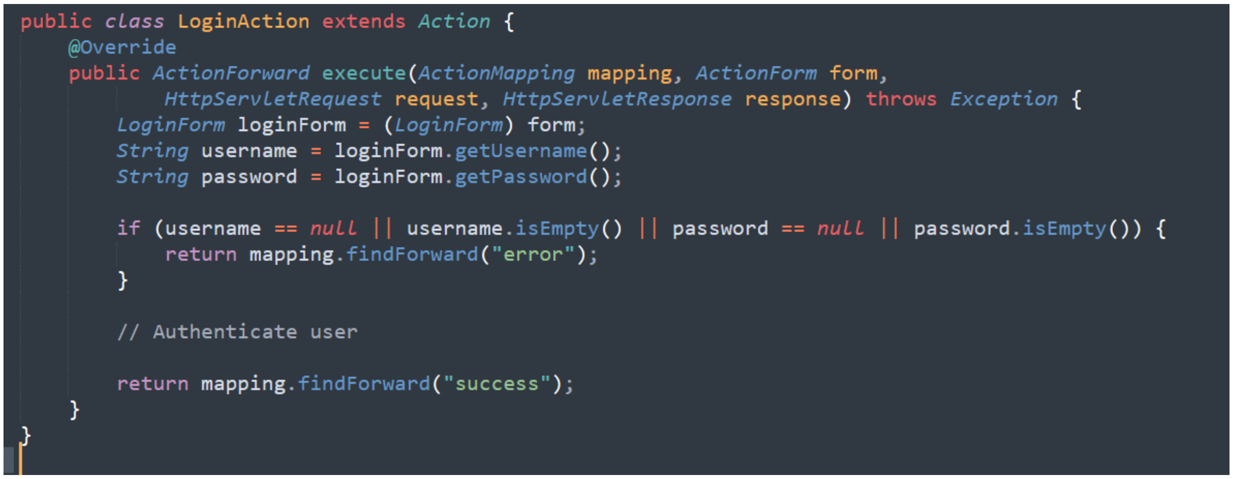
3.2 Form Management

It provides easy form management by using form objects to manipulate HTML forms.

```
“”””

public class LoginForm extends ActionForm {

    private String username;

    private String password;

    // getters and setters

} “”””
```

3.3 Data Validation

Provides validation operations for validating and processing form data



```
public class LoginAction extends Action {
    @Override
    public ActionForward execute(ActionMapping mapping, ActionForm form,
            HttpServletRequest request, HttpServletResponse response) throws Exception {
        LoginForm loginForm = (LoginForm) form;
        String username = loginForm.getUsername();
        String password = loginForm.getPassword();

        if (username == null || username.isEmpty() || password == null || password.isEmpty()) {
            return mapping.findForward("error");
        }

        // Authenticate user

        return mapping.findForward("success");
    }
}
```

4) Apache Wicket

This facilitates web application development by overcoming the low-level complexity of HTTP requests and responses, allowing you to focus on application logic. Wicket applications are built using reusable components, making your code more maintainable and easier to develop. Additionally, Wicket has built-in support for AJAX, a technique aimed at creating dynamic and interactive web experiences.

5. Play Framework

Play Framework is a web application framework that excels in building modern web applications using Java or Scala. Known for its ease of development due to features like leveraging both traditional and functional programming paradigms based on configuration.

Play adopts a lightweight and stateless architecture, making it ideal for building scalable applications. Common use cases for Play include creating RESTful APIs, utilizing websockets, and working with NoSQL and Big Data.

6. Dropwizard

Dropwizard is simplifying the process of creating web services. It focuses on providing developers with the tools they need to build high-performance, reliable, and easily maintainable services. Dropwizard achieves this by offering a collection of pre-integrated libraries that handle common web service functionalities such as configuration, logging, and metrics. This allows developers to concentrate on the core logic of their applications, making it easier to develop, deploy, and monitor web services.

.

7. Junit

It is a popular framework for writing unit tests in Java. However, you may encounter difficulties while testing with JUnit. Solutions related to JUnit address challenges such as organizing large test suites, improving the readability of tests, or automatically obtaining test summaries for better understanding by combining aspects such as these.