

3주차

1. DI와 IoC에 대해 설명해 보세요

IoC는 객체 생성과 관리를 개발자가 아닌 외부 프레임워크나 컨테이너가 담당하는 개념이고, DI는 필요한 객체를 외부에서 주입받는 방식입니다. 둘은 함께 동작해 의존성 관리를 자동화합니다.

2. aop에 대해 설명해주세요

AOP는 공통 기능(예: 로깅, 보안)을 핵심 로직과 분리해 적용하는 방식입니다. Spring에서는 @Aspect와 @Before, @After 같은 애노테이션으로 구현하며, 코드 중복을 줄이고 관리가 쉬워집니다.

3. 클래스와 인스턴스의 차이에 대해 설명해 보세요

클래스는 객체를 만들기 위한 설계도이고, 인스턴스는 그 클래스를 기반으로 생성된 실제 객체입니다. 즉, 클래스는 정의이고 인스턴스는 실체입니다.

4. 변수에 대해 설명하고 let, const의 차이점을 서술하세요.

변수는 값을 저장하고 불러오기 위한 요소입니다. let과 const의 차이점은 변수 재할당 여부에 있습니다. let은 재할당이 가능하고 const는 불가능합니다.

5. get과 post의 차이점은 무엇인가

get은 데이터를 URL에 담아 전송하며 빠르지만 보안에 취약합니다. post는 데이터를 본문에 담아 전송해 보안이 더 좋고, 대용량 전송에 적합합니다.

6. 쿠키와 세션의 차이점은 무엇인가요

쿠키는 클라이언트에 저장되고, 세션은 서버에 저장됩니다. 쿠키는 사용자 정보를 브라우저에 남기고, 세션은 서버에서 사용자 상태를 관리합니다.

7. CORS란 무엇인가요?

CORS는 다른 출처의 리소스를 웹 브라우저가 요청할 때 보안상 제한을 두는 정책입니다. 기본적으로 도메인이 다르면 요청이 차단되는데, 서버가 Access-Control-Allow-Origin 같은 헤더를 통해 허용하면 요청이 가능합니다.

8. 스택과 큐에 대해 설명해주세요

스택은 후입 선출 구조입니다. 마지막 들어온 데이터가 가장 먼저 나가게 됩니다. 큐의 경우 선입선출로 먼저 들어온 데이터가 먼저 나갑니다.

9. HTTP와 HTTPS의 차이점은 무엇인가요?

HTTP는 따로 암호화 과정을 거치지 않기 때문에 중간에 패킷을 가로챌 수 있고, 수정도 가능합니다. 그렇기 때문에 보안이 취약합니다. 이를 보완하기 위해 나온 것이 HTTPS입니다. 중간에 암호화 계층을 거쳐서 패킷을 암호화합니다.

10. OOP 정의는 무엇인가요?

OOP는 객체 지향 프로그래밍을 의미합니다. 데이터를 추상화시켜서 상태와 행위를 가진 객체로 만들고, 그 객체 간의 상호 작용을 통해 로직을 구성하는 방법입니다.