

I. 자바 관련 문제

[질문 1] OOP에 대해 설명하시오.

[답]

Object Oriented Programming (객체지향프로그래밍)으로써 객체라는 단위로 모든 처리를 프로그래밍하는 방법이다. 모든 처리는 객체에 대한 요구의 형태로 표현되며, 요구를 받는 객체는 자기 자신 안에 기술되는 처리를 실행한다. 프로그램이 단순화되고 생산성과 신뢰성이 높은 시스템을 구축할 수 있다. (수정 재사용이 용이) 특징으로는 캡슐화 상속, 다형성, 추상화가 있다.

[질문 2] 캡슐화에 대해 설명하시오.

[답]

객체의 속성(data fields)과 기능(method)을 하나로 묶고 실제 구현 내용 일부를 외부에 감추어 은닉한다.

1. Data 보호

멤버변수로의 직접적인 접근을 막아서 데이터의 안정성을 보장. 데이터를 담고 있는 변수를 private로 선언해서 외부의 접근 차단. Setter/getter 구현한다.

2. 유지보수성

속성이 변경되어도 메소드를 수정하는 것으로 단순하게 처리할 수 있다.

3. 사용자 편의성

내부구현이 달라져도 사용자의 사용법은 변하지 않는다.

[질문 3] 상속에 대해 설명하시오.

[답]

상위객체의 메소드를 그대로 물려받아 재사용 혹은 재정의하면서 새로운 객체를 만드는 방법.

Has a : 하위객체가 상위객체의 모든 것을 물려받음.(부품으로 물려받는 개념)

Is a : 클래스 타입 자체를 물려받으면서 확장하는 방식.

[질문 4] 추상 클래스와 인터페이스에 대하여 설명하시오.

[답]

추상클래스

구체적 클래스를 설계하기에 앞서 필요한 멤버를 결정하고 미구현의 필수 메서드를 정의하여 실제 클래스 구현에 설계도의 역할하는 클래스. 또한 다형성을 통해 프로그램 구현의 효율성을 높일 수 있다. 하위 클래스의 일반화를 통해 구현한다.

인터페이스

멤버변수 선언만으로 이루어지며 클래스와는 구별된다. 클래스를 구현하기에 앞서 기능적인 부문에 일반화를 통해 다형성을 구현하여 상속과는 달리 하나의 클래스에서 여러 인터페이스를 implements를 할 수 있기 때문에 클래스에 필요한 기능의 확장성을 높이고 표준화를 하는데 목적이 있다.

[질문 5] 자바 데이터 타입인 Primitive Type과 Reference Type에 대해 설명하시오.

[답]

Primitive Type은 기본변수로서 변수에 값 자체를 저장하며 Reference Type은 참조형 자료형으로 메모리상에 객체가 있는 위치를 저장한다. Primitive Type의 종류는 boolean, byte, char, short, int, float, long, double 총 8가지이며 Reference Type 종류는 클래스타입, 인터페이스타입, 배열타입, 열거타입이 있다.

[질문 6] 다형성에 대하여 설명하시오.

[답]

하나의 클래스가 여러종류로 정의가능하고 하나의 참조변수로 여러 타입의 인스턴스등의 데이터를 다룰 수 있는 특성을 뜻한다. 조상클래스의 인스턴스를 이용하여 자손타입의 클래스를 다룬다거나, 메서드 오버로딩을 통하여 동일 이름의 메서드를 이용하여 다양한 형태의 파라미터를 다루는 것을 뜻한다.

[질문 7] Thread(스레드)를 구현하는 두 가지 방법에 대하여 설명하시오.

[답]

1. Thread 클래스를 상속 받은 클래스를 제작한 후 run() 함수를 Override하여 구현한다.
2. Runnable 인터페이스를 구현한 클래스를 제작한 후 run() 함수를 Override하여 구현한다.

[질문 8] 컬렉션의 종류 3가지와 그들의 특징에 대하여 설명하시오.

[답]

1. List

순서가 있는 데이터의 집합으로 데이터의 중복을 허용한다.

구현클래스 - ArrayList, LinkedList, Stack, Vector

2. Set

순서를 유지하지 않는 데이터의 집합으로 데이터의 중복을 허용하지 않는다.

구현클래스 - HashSet, TreeSet

3. Map

키와 값의 쌍으로 이루어진 데이터의 집합으로 순서는 유지되지 않으며, 키는 중복을 허용하지 않고 값은 중복을 허용한다.

구현클래스 - HashMap, TreeMap, Hashtable, Properties

[질문 9] 접근 지정자의 종류와 의미에 대하여 설명하시오.

[답]

1. private - 같은 클래스 내에서만 접근이 가능하다.
2. default - 같은 패키지 내에서만 접근이 가능하다.
3. protected - 같은 패키지 내에서, 그리고 다른 패키지의 자손클래스에서 접근이 가능하다.
4. public - 접근 제한이 전혀없다.

[질문 10] Overload와 Override에 대하여 설명하시오

[답]

1. Overload

같은 클래스 안에서 같은 이름의 메소드를 여러 개 가지면서 매개변수의 유형과 개수가 다르도록 하는 기술.

2. Override

상속 관계에 있는 클래스에서 상위 클래스가 가지고 있는 메소드를 하위 클래스가 재정의 해서 사용하는 기술. 상속함으로 통해서 상위클래스의 메소드를 다른 방식으로 사용하기 위해서 재정의하는 것.

II. 데이터베이스 관련 문제

[질문 1] 기본 키(Primary Key)에 대해 설명하시오.

[답]

각 레코드를 구별하기 위해 해당 컬럼 값은 반드시 존재(not null)하고 유일(unique)해야 한다.

[질문 2] 외래 키(Foreign Key)에 대해 설명하시오.

[답]

해당 컬럼 값은 참조되는 테이블의 기본키 값 중의 하나와 일치하거나 null을 가질 수 있으며, 데이터의 무결성을 보장해준다.

[질문 3] DBMS에 대해 설명하시오.

[답]

DBMS란 DataBase Management System의 약자로, 데이터베이스를 구성하고, 이를 효율적으로 응용하기 위해 구성된 소프트웨어를 지칭합니다. 즉, 데이터와 응용 프로그램의 중간에서 응용 프로그램이 요구하는 대로 데이터를 정의하고, 읽고, 쓰고, 갱신하는 등의 데이터를 조작하고 관리하는 프로그램의 집합체가 DBMS이다.

[질문 4] NoSQL의 장단점에 대해 설명하시오.

[답]

1. 장점 : 많은 서버로 확장(Scale-out)이 가능하고 가변적인 구조로 데이터를 저장할 수 있다.
2. 단점 : 다양하고 복잡한 데이터 쿼리는 불가능하며 이터 일관성이 항상 보장되지 않는다.

[질문 5] Join(조인)의 의미와 종류에 대해 설명하시오.

[답]

DBMS의 중요한 목적 중 하나는 데이터를 효율적으로 관리하는 것이다. 그래서 개별 테이블을 만들어 가급적 중복성을 피해서 꼭 필요한 데이터를 저장하게 된다. 하지만 각각의 테이블 하나만 놓고 본다면 그 안에 들어있는 데이터는 정보로서의 가치가 많이 떨어진다. 하지만 각각의 테이블에 분리되어 있는 연관성 있는 데이터들을 연결하거나 조합하여 데이터를 사용할 수 있다면 중요한 정보로서 가공 될 수 있다. 이런 작업들을 조인(Join)이라고 한다.

종류

1. Inner Join

가장 많이 사용되는 방법으로 조인대상이 되는 두 테이블에서 공통적으로 존재하는 컬럼을 연결하여 결과를 생성 하는 조인 방법

2. Outer Join

조인 조건에 만족하지 않는 행들도 나타낼 수 있는 조인이다.

3. Self Join

하나의 테이블에서 서로 조인을 하는 방법이다.

[질문 6] Where 절과 Having 절에 대하여 설명하시오.

[답]

1. Where

Where절은 검색의 결과에 대한 필터링을 해주는 역할을 한다.

2. Having

Having절은 GROUP BY 절에 의해 나올 수 있는 결과에 대한 필터링을 해주는 역할을 한다.

[질문 7] Transaction(트랜잭션)에 대하여 설명하시오.

[답]

트랜잭션은 "하나의 논리적 작업 단위로 수행되는 일련의 작업"로 정의할 수 있다. 이 의미에는 "일련의 작업이 전부 되거나, 전부 되지 않거나"의 의미가 포함되어 있다.

트랜잭션은 ACID라고 불리는 특성을 가지고 있다.

1. Atomicity (원자성)

트랜잭션은 분리할 수 없는 하나의 단위이다. 작업이 모두 수행되거나 하나도 수행되지 않아야 한다.

2. Consistency (일관성)

트랜잭션에서 사용되는 모든 데이터는 일관되어야 한다.

3. Isolation (격리성)

현재 트랜잭션이 접근하고 있는 데이터는 다른 트랜잭션으로부터 격리되어야 한다는 것을 의미한다.

4. Durability (영속성)

트랜잭션이 정상적으로 종료되면, 그 결과는 시스템 오류가 발생하더라도 시스템에 영구적으로 적용되어야 한다.

[질문 8] Commit과 Rollback에 대하여 설명하시오.

[답]

1. Commit

저장되지 않은 모든 데이터를 데이터베이스에 저장하고 현재의 트랜잭션을 종료하라는 명령

2. Rollback

롤백은 로그 파일 중에서 아직 커밋이 되지 않은(즉, 데이터 파일에 적용되지 않은) 내용을 로그 파일의 처음이나 마지막 체크포인트가 설정된 지점까지 취소/삭제시키는 과정을 말한다.

[질문 9] index에 대하여 설명하시오.

[답]

인덱스는 테이블이나 클러스트에서 쓰여지는 선택적인 객체로서, 데이터베이스 테이블내의 원하는 레코드를 빠르게 찾아갈 수 있도록 만들어진 데이터 구조이다.

[질문 10] View에 대하여 설명하시오.

[답]

가상의 테이블이라 하며 view에는 데이터가 들어있는 것이 아닌 SQL만 저장되어 있다. 사용자가 해당 view에 접근하면 그때 view에 들어 있던 SQL이 수행되어 결과를 가져오는 것이다. 일반 테이블을 사용하면 될 텐데 view를 사용하는 이유는 보안과 사용자의 편의성 때문이다. 만약 어떤 테이블에 대해 다른 사용자가 봐서는 안 되는 내용이 있다면 view를 사용하여 정의된 쿼리의 결과만 볼 수 있게 할 수 있다.

III. HTML, JavaScript, CSS, JQuery. Ajax 관련 문제

[질문 1] HTML 언어는 무엇을 목적으로 하는 언어인지를 설명하시오.

[답]

Hyper Text Markup Language(HTML)는 웹페이지를 만들고 보여주는데 사용된다.

문서에 "마크"를 추가하여 작성하는 것이며 오늘날의 "월드 와이드 웹(WWW)"을 특징하는 웹 페이지를 만들고 서로 연결해주는 기능을 가진 언어이다.

한마디로 HTML은 웹 문서의 구조와 의미를 기술하는 언어입니다.

[질문 2] HTML5의 semantic 태그에 대하여 설명하시오.

[답]

기존 HTML4에서는 <div>태그를 이용해 화면을 분할하거나 필요한 것들을 모두 넣었지만 HTML5 부터 모든 태그에 자신의 역할이 생겼다. 이 태그들을 semantic 태그라 한다. semantic 태그는 "의미가 통하는"의 뜻처럼 문서 구조의 가독성을 높여주기 위한 태그이다.

[질문 3] JavaScript의 DOM은 무엇인지 설명하시오.

[답]

문서 객체 모델(Document Object Model, DOM)은 HTML 및 XML 문서를 처리하는 API이다. 문서의 구조적 형태를 제공하므로 자바스크립트(Javascript)와 같은 스크립트 언어를 사용하여 문서 내용과 시각적 표현을 수정할 수 있다.

[질문 4] JavaScript에서 전역 변수와 지역 변수를 선언하는 방법에 대하여 설명하시오.

[답]

1. 전역 변수

함수 밖에서 만들어지는 변수와 함수 안에서 만들어지는 변수라도 var를 이용하지 않은 변수는 전역 변수가 된다.

2. 지역 변수

함수 안에서 만들어지는 변수로 var를 이용해서 만들어진 변수이다.

[질문 5] CSS를 HTML 문서에서 사용하는 방식에 대하여 설명하시오.

[답]

1.외부 스타일 시트(External Style Sheet)

"css"라는 확장자를 가진 스타일 시트 파일을 만들고 이 파일을 HTML 문서에 연결하여 사용하는 방법이다.

2.내부 스타일 시트(Internal Style Sheet)

HTML문서내에서 <head>와 </head>사이에 스타일을 정의하는 방법이다.

3.HTML태그내에 스타일 지정(Inline Styles)

위의 방법들에 비해서 적용범위가 더욱 좁아진 형태로 스타일을 적용하고 싶은 HTML태그 안에서 정의하는 방법이다.

[질문 6] 반응형 웹에 대하여 설명하시오.

[답]

디바이스 종류에 따라 웹페이지의 크기가 자동적으로 재조정 되는 것을 말한다. 어떠한 환경에서도 그에 맞게 사이즈가 변화되어 사용자가 보기 편리하게 만드는 웹이다. 오직 하나의 HTML 소스만으로 특정 장치에 최적화된 환경을 사용자에게 제공할 수 있다.

[질문 7] jQuery에 대하여 설명하시오.

[답]

jQuery는 JavaScript를 좀 더 쉽게 사용하도록 만들어진 JavaScript Library이다. 특징을 살펴보면 다음과 같다.

1. 가벼움

웹에서 가볍다고 하면 용량과도 관계있다. JQuery의 현재 최신 버전인 jquery-1.9.1.min.js 경우 90.4KB 정도의 파일 크기를 가지고 있고 웹에서 동작할 때는 32KB 정도 사용한다.

2. CSS 셀렉터

기존에 CSS는 스타일 파일을 통해서 구현이 가능했었다. JQuery에서는 CSS를 간단한 코드로 변경 가능하도록 적용해서 훨씬 쉽게 사용할 수 있도록 하였다.

3. 크로스 브라우저

최근 웹브라우저는 Internet Explorer, Chrome, Firefox, Safari 등이 다양하게 사용되고 있다. 하지만 JavaScript 로 만든다면 어떤 브라우저에서는 동작하고 다른 브라우저에서는 동작하지 않는 문제가 있다. 그런 문제를 JQuery 에서는 간단하게 지원한다. JQuery 코드로 이벤트를 구현하면 브라우저에 상관없이 지원하게 된다.

[질문 8] jQuery 중에서 Selector에 대하여 설명하시오.

[답]

셀렉터는 jQuery의 가장 강력한 기능이다. HTML 문서 안에는 다양한 엘리먼트들이 포함되어 있는데, 이를 컨트롤하기란 결코 간단한 일이 아니다. 일반적으로 자바스크립트에서 엘리먼트를 얻기 위해 많이 쓰이는 방법은 document.getElementById("엘리먼트 ID 속성 값") 함수를 사용하면 되지만, 일단 소스 코드가 길어지고 얻을 수 있는 엘리먼트들도 매우 한정적이다.

하지만 jQuery에서는 \$("셀렉터 문법") 함수를 사용하여 매우 간단하고 다양한 스타일의 엘리먼트들을 쉽게 얻을 수 있다. 셀렉터로 얻은 엘리먼트들을 확장 집합이라 하는데, 특수한 배열 형태의 객체로 반환이 된다. 이렇게 반환된 객체들은 jQuery가 지원하는 매우 다양한 함수들을 사용할 수 있다.

[질문 9] Ajax에 대하여 설명하시오.

[답]

Ajax는 Asynchronous JavaScript and XML의 약자로서 자바스크립트를 이용해서 비동기적으로 서버와 브라우저가 데이터를 주고받는 방식을 의미한다. Ajax는 웹브라우저와 웹서버가 내부적으로 데이터 통신을 하게 된다. 그리고 변경된 결과를 웹 페이지에 프로그래밍적으로 반영함으로써 웹페이지의 로딩 없이 서비스를 사용할 수 있게 한다.

[질문 10] JSON에 대하여 설명하시오.

[답]

JSON(JavaScript Object Notation)의 약자로 JavaScript에서 객체를 만들 때 사용하는 표현식을 의미한다. 이 표현식은 사람도 이해하기 쉽고 기계도 이해하기 쉬우면서 데이터의 용량이 작다. 이런 이유로 최근에는 JSON이 XML을 대체해서 설정의 저장이나 데이터를 전송 등에 많이 사용된다.

IV. Servlet/JSP/Spring, MyBatis 관련 문제

[질문 1] DI에 대하여 설명하시오.

[답]

Dependency Injection의 약자로 의존성을 주입하는 것을 뜻한다. 설정 파일을 통해 객체간의 의존관계를 설정함으로써 외부 Assembler가 객체간의 의존 관계를 정의하게 되며, 객체는 직접 의존하고 있는 객체를 생성하거나 검색할 필요가 없어지므로 코드의 관리가 쉬워지고 확장성이 높아진다.

[질문 2] AOP에 대하여 설명하시오.

[답]

Aspect Oriented Programming을 뜻하며 다양한 곳에서 자주 사용되는 공통 관심요소를 단일 기능으로 뽑아내어 코드의 중복을 줄이고 관리의 효율성을 높이는 것을 목적으로 한다. 로깅이나 로그인등의 기능을 예로 들 수 있다.

[질문 3] Was에 대하여 설명하시오.

[답]

Web Application Server의 약어로 웹과 기업의 기간 시스템 사이에 위치하면서, 웹 기반 분산 시스템 개발을 쉽게 도와주고 안정적인 트랜잭션 처리를 보장해 주는 일종의 미들웨어 소프트웨어 서버를 말한다.

[질문 4] MVC에 대하여 설명하시오.

[답]

MVC란 Model View Controller의 약자로 애플리케이션을 세 가지의 역할로 구분한 개발 방법론이다. 사용자가 Controller를 조작하면 Controller는 Model을 통해서 데이터를 가져오고 그 정보를 바탕으로 시각적인 표현을 담당하는 View를 제어해서 사용자에게 전달하게 된다.

1. Controller : 사용자가 접근 한 URL에 따라서 사용자의 요청사항을 파악한 후에 그 요청에 맞는 데이터를 Model에 의뢰하고, 데이터를 View에 반영해서 사용자에게 알려준다.
2. Model : 일반적으로 CI의 모델은 데이터베이스 테이블에 대응된다.
3. View : View는 클라이언트 측 기술인 html/css/javascript들을 모아둔 컨테이너이다.

[질문 5] Session(세션)과 Cookie(쿠키)에 대하여 설명하시오.

[답]

1. 쿠키

쿠키란 서버측에서 클라이언트측에 상태 정보를 저장하고 추출할 수 있는 메커니즘으로 클라이언트의 매 요청마다 웹 브라우저로부터 서버에게 전송되는 정보패킷의 일종이다. HTTP에서 클라이언트의 상태 정보를 클라이언트의 하드 디스크에 저장하였다가 필요 시 정보를 참조하거나 재사용할 수 있다.

2. 세션

세션이란 클라이언트와 웹서버 간에 네트워크 연결이 지속적으로 유지되고 있는 상태를 말한다. 클라이언트가 웹서버에 요청하여 처음 접속하면 JSP(혹은 ASP)엔진은 요청한 클라이언트에 대하여 유일한 ID를 부여하게 되는데, 이 ID를 세션이라 부르며 세션 ID를 임시로 저장하여 페이지 이동 시 이용하거나, 클라이언트가 재접속 했을 때 클라이언트를 구분할 수 있는 유일한 수단이 된다

3. 쿠키와 세션의 차이점

쿠키(cookie)와 세션(session)은 기능상 비슷한 역할을 하고, 동작원리도 비슷하다. 왜냐하면, 일반적인 세션은 쿠키를 바탕으로 동작하기 때문이다. 그러나 가장 중요한 차이점은 저장되는 곳이 다르다는 것이다. 쿠키는 클라이언트에 저장되고, 세션은 서버에 저장된다. 쿠키의 경우에는 서버의 자원을 전혀 사용하지 않지만, 세션의 경우에는 서버에 저장되기 때문에 서버의 자원을 사용할 수가 있다. 쿠키와 세션의 만료 되는 기간도 다르다.

[질문 6] Servlet의 생명주기에 대하여 설명하시오.

[답]

1. 클라이언트가 최초로 요청할 때 서블릿이 생성되면서 한번 init()메소드를 호출한다.
2. 클라이언트의 요청이 있을 때마다 service 메소드가 호출된다.
3. 일정 시간 클라이언트의 요청이 없거나 서버가 정상적인 방법으로 종료되면 서블릿이 소멸되면 destroy() 메소드를 호출한다.

[질문 7] Model1과 Model2의 차이점에 대하여 설명하시오.

[답]

1. Model1

JSP 파일내에서 뷰와 컨트롤러 모두 실행되어 지는 구조.

장점 : 개발 기간 단축(단순한 페이지 흐름)

팀원의 수준이 높지 않아도 구현 가능. 초보자도 쉽게 구현.

중소형 프로젝트에 적합.

단점: 웹 어플리케이션이 복잡해질수록 유지보수가 어렵다.

디자이너와 개발자간의 의사소통이 필요하다.

(비즈니스 로직과 뷰 사이의 구분이 미비하다.)

2. Model2

Controller(servlet)을 통해 비즈니스 로직과 뷰가 분리된 형태

JSP에서는 뷰를 담당하고, 자바 클래스에서 비즈니스 로직을 담당한다.

장점: 비즈니스 로직과 뷰의 분리로 유지보수와 확장이 용이하다.

개발자와 디자이너의 작업이 분리되어 분업이 편리하다.

단점: 구조 설계를 위한 시간이 많이 소요되므로 개발 기간이 증가한다.

개발자들이 구조에 대한 이해가 필요하기 때문에 팀원의 높은 수준이 요구된다.

[질문 8] Connection Pool에 대하여 설명하시오.

[답]

DB Connection Pool이란 DB Connection Pool 매니저가 일정한 커넥션을 연결하고 있다가, 요청이 들어오면 커넥션을 할당을 해주고 없으면 기다리게 한다. 요청한 클라이언트는 커넥션을 다 사용하면 다시 반납하는 구조로 이루어진다. 따라서 속도 면이나 퍼포먼스 부분에서 조금은 향상을 바랄 수 있다.

[질문 9] MyBatis에서 resultType과 parameterType에 대하여 설명하시오.

[답]

1. resultType

질의 결과를 기억할 데이터 형태를 말한다.

2. parameterType

질의를 실행하기 위해 제공해야 할 데이터 형태를 말한다.

[질문 10] DAO와 Service의 역할에 대하여 설명하시오.

[답]

1. DAO

Data Access Objects의 약어로써 실질적인 DB와의 연결을 담당하는 일을 가진 객체를 말합니다. DAO를 사용하는 이유는 효율적인 커넥션 관리와 보안성입니다.

2. Service

Controller에서 사용하는 비즈니스 로직이 포함된 객체를 말한다.

여러 DAO를 호출하여 여러번의 데이터 접근/갱신을 하며 그렇게 읽은 데이터에 대한 비즈니스 로직을 수행하고, 그것을 하나의(혹은 여러개의) 트랜잭션으로 묶습니다. 즉, Service가 트랜잭션 단위이다.