

면접 예상 질문과 답변지

중앙정보 취업지원센터

Tel : 02-712-0524

Fax : 02-393-9218

Mobile :

- 홍성표 컨설턴트 : 010-5072-2311
- 심원태 컨설턴트 : 010-9276-3769

e-mail : job@choongang.co.kr

- ※ 본인의 생각을 진솔하게 말씀하시되 약간은 보수적으로 말씀하실 필요가 있습니다.
- ※ 자신감을 가지고 면접관의 질문에 큰소리로, 간단, 명료하게 답변합니다.

1. 자기소개를 2~3 분 동안 해 보세요.
 - > 반복연습을 통해 숙지, 100% 암기는 NO! 자연스럽게, 자신감 있는 태도 중시
2. 자기 성격의 장점(강점)은 무엇인가요?
3. 자신의 단점이 무엇이고 어떻게 보완하고 있는지?
4. (이력서에 비어있는 기간) 그 기간 동안 무엇을 했나요?
 - > 공부, 향후 할 일에 대한 준비 기간으로 활용
5. 야근을 하라고 하면 어떻게 할 것인가요?
 - > 배울 기회가 왔으므로 기쁘게 참석하겠다는 의지 표현
6. 상사의 의견과 다른 경우에는 어떻게 대응할 것입니까?
 - > 상사의 의견을 따르고 사석에서 의견을 전달해 보겠다.
7. 왜 우리회사에 지원 했지요?
 - > 회사 홈페이지에서 정보를 파악하고 회사의 주요 업무와 일치하는 부분과 연계하여 답변
8. 우리 회사가 무엇을 하는 회사로 알고 있나요?
 - > 홈페이지를 통하여 사전에 회사 환경을 파악하고 답할 것
9. 우리 회사에 대하여 궁금한 것이 있으면 질문하세요.
 - > 가능한 업무 쪽으로 질문하고 근무환경은 가볍게 질문
10. 본인이 한 프로젝트(교육 포함)을 설명하고 역할을 대하여 말해 보세요.
 - > 목적, 프로젝트 구성, 업무 또는 데이터 흐름 등
11. 다른 회사에도 지원했을 텐데 만약, 우리 회사를 선택하지 않고 다른 회사를 선택했다면 그 이유가 무엇인지? (지망동기)
12. 원하는 연봉 수준은 얼마인가?
13. 어느 부서에서 일하고 싶은가? 그 이유는? (직업관)
14. 개발자로 전직한 이유는 무엇이었는지?
 - 1) 컴퓨터에 흥미가 많았고 적성이 맞음
 - 2) IT분야는 끝없이 발전하고 전망이 밝다고 생각한다.
 - 3) 성격이 끈기가 있고 체계적이고 논리적으로 하는 업무가 좋음
15. 인생의 꿈과 목표에 대해 말해 보시오.
16. 지금까지 살아오면서 가장 어려웠던 일과 즐거웠던 일은?
17. 주말에는 주로 무엇을 하며 시간을 보내나?
18. 대학시절 학업 외에 가장 즐겨 했던 일은? (대학생활, 친구)
19. 오늘 조간신문의 톱기사는 무엇인가? (일상적인 인생관)
20. 최근에 읽은 책이 있다면 내용과 느낀 소감을 말해보라. (일상적인 인생관)
21. 자소서 궁금사항 질문
 - > 자소서 작성 시 면접을 고려하여 작성하여야 한다.(충분한 설명이 될 수 있도록 준비)
22. 팀 프로젝트 중 자신이 생각하는 자신의 기여도와 그 이유
23. 같이 들었던 수강생 중 자신의 실력을 몇 등이라고 생각하는지 또 그렇게 생각한 이유
24. 스트레스 해소는 어떻게 하는지?

25. 대학교 졸업 후 2 년의 공백기에 대한 질문
26. 이론 테스트 (꼭지 시험)
27. 현재 다루는 프로그래밍 언어에는 무엇이 있는가?
28. 실제로 프로그래밍 작업에 참여해 본 적이 있는가?
29. 프로그래밍 설계와 코딩에 대해 본인이 직접 구현한 것이 있는가?
30. 개발자 커뮤니티 등 IT 관련 동호회 활동을 하는가?
31. 개발자로서 개인적으로 어떤 로드맵을 갖고 있는가?
32. 관심 있는 IT 분야가 어디인가?
33. 관심 있는 산업분야가 어디인가?
34. 친구들이 많이 있는가? 선후배들이 많은가? 사이가 좋은가?
35. 스트레스가 쌓이거나 힘든 일이 생기면 어떻게 극복하나?
36. 취미가 무엇인가?

- 코딩테스트

- for 문
- 클래스 정의
- strip, rstrip, lstrip
- split
- while 문
- list 인덱싱

■ 1분 자기소개

<1분 자기소개 7가지 유의사항>

1. 자신감 있는 태도 중시
2. 직무 역량 강조, 강점은 구체적으로
3. 좋은 내용 많이 전달 (하나의 메시지 집중 전달)
4. 귀에 쏙쏙 들어오도록 내용 강조
5. 반복 연습을 통한 내용 숙지
6. 100% 암기는 NO! 자연스럽게
7. 시간 준수는 기본!

<1분 자기소개 사례1>

안녕하십니까? 탄탄한 기본기를 갖춘 웹 개발 지원자 000입니다.

저는 직업훈련기관인 중앙정보처리학원에서 <???>과정을 6개월 960시간 수료하였습니다.

수료한 교육 내용은

Java, JSP, Spring..... 를 배웠습니다.

또한 팀 프로젝트를 통해 실제 현장에 투입되더라도

바로 실무를 수행 할 수 있는 실력을 갖추고 있습니다.

.....

6개월의 교육과정을 통한 탄탄한 실무지식을 바탕으로 저의 개발능력을 귀사에서 발휘하고 싶습니다.

귀사에서 저와 같은 성실하고, 능력 있는 인재를 채용 하시기를 바랍니다.

<1분 자기소개 사례2>

“경로 안내를 시작하겠습니다. 목적지는 00회사입니다”

안녕하십니까? 인간 내비게이션 지원자 000입니다

내비게이션이 경로를 벗어났다고 목적지 안내를 포기하는 걸 보신 적 있으십니까?

저 또한 전원만 켜져 있다면, 살아만 있다면 목표를 끝까지 달성하는 사람입니다.

00회사를 소프트웨어의 글로벌 전문기업으로 성장할 수 있도록

가장 신속하고 정확하게 인도하겠습니다

<효과적인 자기소개 리뷰>

- 자기소개 좋은 신호
면접관이 고개를 들고 지원자를 바라본다면?
추가질문을 받는다면?
- 자기소개 전략
A-B-A' 내용 구성(A-강조 포인트 꺼내기, B-내용 자세하게 말하기, A'-강조 포인트 요약 정리하기)
자신감 있는 태도 등 7가지 유의사항 체크
사례 벤치마킹, 개성과 진정성 필수

[Programming]

1. 프로그램이 무엇인지 말해 보세요.

- 컴퓨터가 사람 일을 할 수 있도록 해 주는 것
- 컴퓨터에 처리되는 작업의 순서를 논리적으로 명령어로 작성하는 것

2. JAVA언어의 좋은 점에 대하여 말해 보세요.

객체지향형 프로그래밍 언어, 플랫폼 독립적, 이식성이 좋다, 라이브러리 지향성, 보안성, 멀티 스레드, 가상머신, 바이트 코드

3. 객체지향 언어(Object Oriented Language)의 장점과 객체지향 프로그램이 등장한 이유에 대해 설명하세요.

- 코드의 재사용성이 높아 새로운 코드를 작성할 때 기존의 코드를 이용하여 쉽게 작성할 수 있다.
- 코드의 관리가 용이하여 코드간의 관계를 이용해서 적은 노력으로 쉽게 코드를 변경할 수 있다.
- 제어자와 메서드를 이용해서 데이터를 보호하고 올바른 값을 유지하도록 하며, 코드의 중복을 제거하여 코드의 불일치로 인한 오 동작을 방지할 수 있다.

4. 자바의 데이터 타입인 Primitive Type 과 Reference Type 에 대해 설명하세요.

- Primitive Type 은 변수에 값 자체를 저장하며 Reference Type 은 메모리상에 객체가 있는 위치를 저장한다.
- Primitive Type의 종류는 boolean, byte, char, short, int, float, long, double 총 8가지 이며 Reference Type 종류는 클래스타입, 인터페이스타입, 배열타입, 열거타입이 있다.

5. 배열과 컬렉션의 차이점에 대해서 설명하세요.

- 배열은 단 하나의 자료형만 저장이 가능하고, 컬렉션은 복수의 자료형을 저장할 수 있다.
- 배열은 고정된 크기이고, 컬렉션은 가변적 크기이다.

6. 다형성이란 무엇인가?

- 여러 가지 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.

7. 멀티 스레드의 장단점은 무엇인가?

두 가지 이상의 작업을 동시해 실행 할 수 있어 자원을 효율적으로 이용할 수 있으나 dead lock 및 동기화에 대한 철저한 검증이 필요하다.

8. 자바에서 멀티스레드를 구현하는 방법에 대해서 설명하세요.

Thread 클래스를 상속하는 법. 단일 상속만 된다는 단점이 있다. Runnable 인터페이스를 상속하는 법. 다중 상속이 된다는 장점이 있다.

9. Java 컬렉션의 대표 인터페이스는 무엇인가?

List

순서가 있는 데이터의 집합으로 데이터의 중복을 허용한다.

구현클래스 - ArrayList, LinkedList, Stack, Vector

Set

순서를 유지하지 않는 데이터의 집합으로 데이터의 중복을 허용하지 않는다.

구현클래스 - HashSet, TreeSet

Map

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

구현클래스 - HashMap, TreeMap, Hashtable, Properties

10. 컬렉션에서 제네릭이 추가된 이유에 대해서 설명하세요.

컬렉션은 복수개의 데이터 타입 요소값이 저장되다 보니 원하는 자료형 타입 요소값을 추출하기 어렵다. 그러므로 지정한 자료형 타입 한가지만 저장하기 위해서 나온 것이 제네릭이다.

11. 접근제어자의 종류와 특성에 대하여 설명하라.

private – 같은 클래스 내에서만 접근이 가능하다.

default – 같은 패키지 내에서만 접근이 가능하다.

protected – 같은 패키지 내에서, 그리고 다른 패키지의 자손클래스에서 접근이 가능하다.

public – 접근 제한이 없다.

접근 허용 범위는 다음과 같다 public > protected > default > private

12. Wrapper 클래스란 무엇인가?

primitive 타입으로 표현할 수 있는 간단한 데이터를 객체로 만들어야 할 경우가 있는데 그러한 기능을 지원하는 클래스를 뜻하며 각 primitive 타입에 대응하는 Wrapper 클래스는 아래와 같다.

byte => Byte
short => Short
int => Integer
long => Long
char => Character
float => Float
double => Double
boolean => Boolean

13. 추상클래스란 무엇인가?

클래스를 설계도에 비유한다면 추상클래스는 미완성 설계도에 비유할 수 있다. 미완성 설계도란, 단어의 뜻 그대로 완성되지 못한 채로 남겨진 설계도를 말한다. 클래스가 미완성이라는 것은 멤버의 개수에 관계된 것이 아니라, 단지 미완성 메서드(추상메서드)들 포함하고 있다는 의미이다. 미완성 설계도로 완성된 제품을 만들 수 없듯이 추상클래스로 인스턴스는 생성할 수 없다. 추상클래스는 상속을 통해서 자손클래스에 의해서만 완성될 수 있다.

14. 인터페이스란 무엇인가?

인터페이스는 일종의 추상클래스이다. 인터페이스는 추상클래스처럼 추상메서드를 갖지만 추상클래스보다 추상화 정도가 높아서 추상클래스와는 달리 몸통을 갖춘 일반 메서드 또는 멤버변수를 구성원으로 가질 수 없다. 오직 추상메서드와 상수만을 멤버로 가질 수 있으며, 그 외의 어떠한 요소도 허용하지 않는다. 추상클래스를 부분적으로만 완성된 '미완성 설계도'라고 한다면, 인터페이스는 구현된 것은 아무것도 없고 밑그림만 그려져 있는 '기본 설계도'라고 할 수 있다. 인터페이스는 인터페이스로부터만 상속받을 수 있으며, 클래스와는 달리 다중상속, 즉 여러 개의 인터페이스로부터 상속을 받는 것이 가능하다.

14-1. 인터페이스를 왜 쓰는지 설명해 보세요.

. 다형성을 구현하기 위해서 사용한다

즉 같은 인터페이스를 구현하는 클래스들은 실행하는 객체가 인터페이스를 통하여 다른 객체를 사용할 수 있도록 하여 클래스간의 결합도를 낮추고 응집도를 높일 수 있음. 예로 스프링에서 DI(의존성 주입) 등에서 다형성을 구현하고 결합도를 낮추기 위해 사용

. 양식 역할 : 규칙에 맞게 구현하도록 표준을 제시하고 클래스들이 메세드를 빠짐없이 재정의 하도록 한다

15. 자바의 GUI에서 스윙과 AWT의 차이점에 대해서 설명하세요!

AWT는 운영체제가 갖고 있는 각 컴포넌트를 이용한다. 즉 운영체제에 종속적인 GUI로서 운영체제 종류에 따라 화면에 출력되는 모양이 약간씩 다르다.

스윙은 운영체제가 갖고 있는 GUI를 사용하지 않고 자바 가상머신의 Swing 패키지를 직접 사용해 구현한다.

따라서 운영체제가 서로 달라도 동일한 화면을 출력하게 한다. 스윙은 AWT보다 세련되고 다양한 GUI를 제공해 주기 때문에 자바 어플 GUI 개발에서는 AWT보다는 스윙을 더 많이 사용한다.

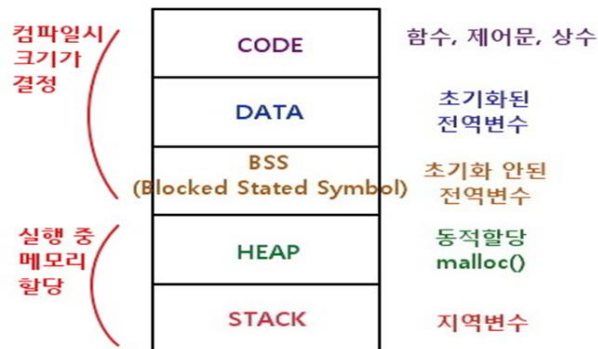
16. 가비지 콜렉터(Garbage Collection)에 대하여 말해 보세요.

쓰레기 수집(Garbage Collection)은 동적 할당된 메모리 영역 가운데 더 이상 사용할 수 없게 된 영역을 탐지하여 자동으로 해제하는 기법이다. 쓰레기 수집이 지원되는 환경에서는 프로그래머가 동적으로 할당한 메모리 영역의 전체를 완벽하게 관리할 필요가 없어진다. 이러한 작업은 CLR의 '가비지 콜렉터'라는 일종의 백그라운드 서비스를 통해 자동으로 이뤄진다

17. 컬렉션 List 인터페이스를 구현한 ArrayList 컬렉션 클래스에 대해서 말해 보세요.

- 가변적 크기이다
- 순차적으로 요소 값을 저장한다.
- 중복 요소 값을 허용한다.

17-1. 자바(JAVA)의 메모리 구조가 어떤 식으로 이루어져 있는지?



17-2. out of memory 에러 메시지는 어느 메모리 부족으로 나타나는 것인지?

- heap 메모리 부족

18. 메서드에서 오버로딩과 오버라이딩(Overriding)에 대하여 말해 보세요.

@오버로딩 : 메소드 이름은 같지만 매개변수의 개수나 데이터 형식을 다르게 정의하는 것

@오버라이딩(Overriding) : 기존에 있는 메소드를 재정의하는 것으로 매개변수의 개수나 데이터형식이 같아야 한다

19. Exception 에 대하여 말해 보세요.

- 프로그램의 비정상적인 종료나 종단을 막을 수 있다.
- RuntimeException 과 그 외의 Exception 으로 나눌 수 있다.

20. OSI 7 계층(Layer)에 대하여 말해 보세요.

▶ OSI (Open System Interconnection) 모델의 7 개 계층구조

| 구 분 | | 설 명 |
|---|--------------------------------|--|
| 네트워크 지원계층 (한 장치에서 다른 장치로 데이터를 이동할 때 필요한 물리적인 면(즉, 전기적인 규격, 물리적인 연결, 물리주소, 전송시간과 신뢰도 등) 처리) | 1. Physical layer (물리층) | 물리적 매체를 통해 비트 흐름을 전송하기 위해 필요한 기능들을 조정하고, 인터페이스의 기계적·전기적 규격, 전송매체를 다룸. 물리적인 장치와 인터페이스가 전송을 위해 필요한 기능과 처리절차를 규정함 |
| | 2. Data link layer (데이터링크층) | 가공되지 않은 내용의 전송을 담당하는 물리층을 신뢰성 있는 링크로 변환시켜 주고 노드-대-노드 전달(node-to-node delivery)함 |
| | 3. Network layer (네트워크층) | 패킷을 발신지로부터 여러 네트워크(링크)를 통해 목적지까지 전달함 |
| 전송층 (종단-대-종단까지의 믿을 만한 데이터 전송 보장) | 4. Transport layer (전송층) | 전체 메시지의 프로세스-대-프로세스 전달을 함 |
| 사용자 지원계층 (서로 상관없는 소프트웨어 시스템 사이의 상호연동을 가능하게 함) | 5. Session layer (세션층) | 네트워크의 대화 조정자로 통신하는 시스템들 사이의 상호작용을 설정·유지하고 동기화 함 |
| | 6. Presentation layer (표현층) | 두 시스템 사이에서 교환되는 정보의 구문과 의미에 관련되어 변환, 압축 및 암호화를 담당함 |
| | 7. Application layer | 사용자(사람 또는 소프트웨어)가 네트워크에 접근할 수 있도록 함. 사용자 인터페이스를 제공하고, 전자우편, 원격 파일 |

| 구 분 | | 설 명 |
|-----|-------|--|
| | (응용층) | 접근과 전송, 공유 데이터베이스 관리 및 여러 종류의 분산 정보 서비스를 제공함 |



21. 동기화(synchronized)에 대하여 설명해 보세요.

- 하나의 자원을 여러 태스크가 사용하려 할 때에, 한 시점에서 하나의 태스크만이 사용할 수 있도록 하는 것

22. 프로세스와 쓰레드의 차이점에 대하여 말해 보세요.

- 프로세스 : 실행 중인 프로그램, 자원(resources)과 쓰레드로 구성
- 쓰레드 : 프로세스 내에서 실제 작업을 수행하는 단위, 모든 프로세스는 하나 이상의 쓰레드를 가지고 있다.
- 다중 쓰레드 : 하나의 프로세스(프로그램)에 하나 이상의 쓰레드를 생성하여 실행

[Database]

1. 데이터베이스 종류에 대하여 말해 보세요.

- 1) Hierarchical DataBase : IMS/DB, HDB
- 2) Network DataBase
- 3) Relational DB, OR(Relational, Object) DB
 - 대형 Oracle, DB2, Sysbase, Informix
 - 소형 MS-SQL, My SQL, PostgreSQL
 - 기타 MongoDB
- 4) OODB(Object Oriented DataBase)

※Oracle과 MySQL의 차이를 말해 보세요.

- Oracle : 대용량처리에 적합, UNIX, Linux, 메인프레임 등에서 사용, DB관리자 별도
- MySQL : 5000만건 미만의 데이터 주로 PC, UNIX시스템에서 사용, 보통 개발자가 DB관리
- SQL의 80% - 90%가 비슷 하지만
 - MySQL에 없는 Oracle 명령어 : varchar2, nvl, nvl2, sequence, decode, outer join에서 + 등
 - Oracle에 없는 MySQL 명령어 : autoincrement, show, ifnull 등

2. 데이터베이스 언어(Database Language)에 대하여 말해 보세요.

- DDL(Data Definition Language) : CREATE, ALTER, DROP
- DML(Data Manipulation Language) : SELECT, INSERT, DELETE, UPDATE
- DCL(Data Control Language) : COMMIT, ROLLBACK, GRANT, REVOKE

3. 각종 제약사항(Constraint)에 대하여 말해 보세요.

- Not Null
- Primary key
- Foreign Key
- Unique
- Default
- Check

4. 검색성능을 향상 시키는 색인(index)에 대하여 말해 보세요.

색인(index)은 컴퓨터에서 내용을 미리 목록으로 만들어 놓고 찾고자 하는 내용을 검색하는데 시간을 줄이기 위한 것이다.

색인은 검색성능을 향상시키기 위한 것으로 정보 요구자가 보다 빨리 정보에 접근할 수 있도록 그 정보의 소재를 표시해 주고, 원하는 자료의 유무를 확인시켜 주며 자료의 신속한 이용을 가능하게 하는 기능을 가지고 있다.

인덱스(색인)이 많으면 조회 속도는 빠르지만 입력/수정/삭제의 속도는 떨어진다

5. 트랜잭션(transaction)에 대하여 말해 보세요.

데이터베이스 트랜잭션(Database Transaction)은 데이터베이스 관리 시스템 또는 유사한 시스템에서 상호작용의 단위로 논리적 작업 단위(LUW, Logical Units of Work)이다. 여기서 유사한 시스템이란 트랜잭션이 성공과 실패가 분명하고 상호 독립적이며, 일관되고 믿을 수 있는 시스템을 의미한다.

데이터베이스 시스템은 각각의 트랜잭션에 대해 원자성(Atomicity), 일관성(Consistency), 고립성(Isolation), 영구성(Durability)을 보증한다. 이 성질을 첫 글자를 따 ACID라 부른다.

Begin transaction

Commit transaction (트랜잭션이 성공적이며, 갱신이 실제 적용됨)

6. 커밋(Commit)과 롤백(Roll back)에 대하여 설명해 보세요

- Commit - 처리결과와 영구적 반영을 시행한다.
- Rollback - 결과를 취소, 트랜잭션의 처음 시점으로 되돌린다

7. CRUD 란 무엇인가?

| 이름 | 조작 | SQL |
|--------------------|-----------|--------|
| Create | 생성 | INSERT |
| Read(또는 Retrieve) | 읽기(또는 인출) | SELECT |
| Update | 갱신 | UPDATE |
| Delete(또는 Destory) | 삭제(또는 파괴) | DELETE |

8. 커서(Cursor)에 대하여 말해 보세요.

커서(Cursor)는 일련의 데이터에 순차적으로 액세스할 때 검색 및 "현재 위치"를 포함하는 데이터 요소이다.



[Servlet & JSP]

1. Server Side Applet인 Servlet에 대하여 정의하고 Servlet클래스를 만들기 위한 방법을 기술하시오.

JAVA Servlet은 JAVA를 사용하여 웹페이지를 동적으로 생성하는 서버측 프로그램 혹은 그 사양을 말하며, 흔히 "서블릿"이라 불린다.

서블릿은 JAVA EE 사양의 일부분으로, 주로 이 기능을 이용하여 쇼핑몰이나 온라인 뱅킹 등의 다양한 웹 시스템이 구현되고 있다.

서블릿은 외부 요청마다 프로세스보다 가벼운 스레드로써 응답하므로 보다 가볍다. 또한, 서블릿은 JAVA로 구현되므로 다양한 플랫폼에서 동작한다.

2. Servlet과 JSP의 차이점에 대하여 말해 보세요.

- Servlet은 JAVA 소스에 HTML코드가 삽입된다.
 - JSP는 반대로 HTML코드에 JAVA코드가 삽입된다.
 - Servlet Class는 컴파일과정과 등록 과정이 필요하지 않지만 JSP는 필요 없다.
 - Servlet 보다는 JSP 디자인과 로직에 대한 구분이 명확해서 유지보수가 용이하다.
 - 간단한 로직을 구현할 때는 JSP가 더 간편하다.
 - 하지만 복잡한 로직을 구현할 때에 HTML 중심의 코드가 이해하기 어렵게 만들 수 있고, 프로그래밍 언어를 모르는 사람이 실수로 중요한 코드를 건들 우려도 있다. 그리고 힘들게 개발한 로직의 유출을 막기 위해서도 Servlet 기술이 필요하게 된다.
- 그래서 요즘은 JSP 기술과 Servlet 기술을 혼용한 MVC 프로그래밍 방법이 권장되고 있다. 프로그램의 기능을 구현하는 복잡한 로직은 서블릿 클래스 안에 기술하고, 그 결과를 가져다가 출력하는 일만 JSP 페이지가 담당하도록 만드는 방법이다.

3. JSP구성요소인 지시어(Directive), 주석문(Comments), 선언문(Declarations), 연산문(Expressions), 수행문(Scriptlets) 을 설명하세요.

* 지시어(Directive) --> <%@ %>

- 1)page : JSP 페이지에 대한 정보를 지정. (문서의 타입, 출력 버퍼 크기, 에러 페이지 등.)
 - language, contentType, import, session, buffer, autoFlush, errorPage, isErrorPage 등등
- 2)tablib : JSP 페이지에서 사용 할 태그 라이브러리 지정
- 3)include : JSP 페이지의 특정 영역에 다른 문서를 포함

* 주석문(Comment)

- 1)HTML <!-- --> : 소스보기로 코드를 볼 수 있다
- 2)JAVA //, /* ~ */ : 소스보기로 코드를 볼 수 없다
- 3)JSP <%-- --%>

* 선언문(Declarations)

<%! %> : JAVA 메서드를 만든다

* 연산문(Expressions)

<%= %> : 값을 출력한다

* 수행문(Scriptlets)

<% %> : 자바 코드를 실행한다.

- Action Tag : <jsp:~ />

- Custom Tag

- EL (Expression Language)

- Implicit Object -- 내장객체

3-1. request와 response 객체에 대해 말해보시오.

request : 클라이언트에서 넘어 오는 데이터를 전달 받기 위한 객체

response : 서버에서 클라이언트로 데이터를 전달하기 위한 객체

3-2. 클라이언트에서 데이터를 넘기는 방법에 대해 설명하시오

- 1) html에서 form 객체를 작성하여 action 으로 정해진 서버의 url로 전달하는데 전달할 때는 submit 함수를 이용해 전달하게 된다.
- 2) get 방식으로 url 뒤에 ?를 이용하여 작성하여 준다.

4. name, age를 담은 javaBean을 작성해 보세요.

```
public class Person {
    private String id;
    private String name;
    public String getId() {
        return id;
    }
    public void setId(String id) {
        this.id = id;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
}
```

5. <jsp:setProperty/><jsp:getProperty/>그리고<jsp:useBean/>을 언제 사용하는 것입니까?

화면에서 입력 받은 데이터를 받아서 저장할 객체를 생성하고(<jsp:useBean/>)
생성된 객체의 멤버변수에 데이터를 저장(<jsp:setProperty/>)
저장된 데이터를 사용하기 위하여 가져오는 경우(<jsp:getProperty/>) 사용

6. Servlet의 LifeCycle에 해당하는 메소드와 그 메소드에서 처리될 내용에 대해 설명하시오.

- * 서블릿은 init(), service(), destroy() 메소드를 호출하는 Life Cycle을 가짐.
- * init() 메소드
 - 서블릿이 메모리에 Load 되면 init() 메소드 수행.
 - 서블릿이 서비스하기 위해 필요한 초기화 작업 수행.
 - 한 번만 수행. (병행적으로 수행되지 않음.)
 - 서블릿이 실행하기 위해서 필요한 각종 환경을 설정하기 위한 목적으로 사용.
(필요한 파일 열기, 데이터베이스에 연결 등)
 - 맨 처음 클라이언트의 요청에 의해서 메모리에 로드되며, 메모리에 로드된 후에는 메모리에 계속 남아 클라이언트의 요청을 처리.
- * service() 메소드
 - 클라이언트의 요청이 있을 때마다 Thread가 생성되어서 병행적으로 service() 메소드 수행.
 - 병행성 문제 고려.
 - HTTP의 method 타입에 따라 GET 방식이면 doGet()메소드를, POST 방식이면 doPost()메소드 호출.
- * destroy() 메소드
 - 메모리가 Unload되기 전에 destroy() 메소드 수행.
 - 한 번만 수행. (병행적으로 수행되지 않음.)
 - 종료 시에 필요한 끝내기에 관련된 작업을 처리.
- * 서블릿 작업 중단
 - 서블릿 작업 종단을 위해서는 doPost(), doGet(), service()등의 함수에서 return문을 사용.
 - System.exit()를 호출하면 서블릿 컨테이너가 종료.

8. HTTP수행방식 중 GET방식과 POST방식의 차이점을 기술하시오.

- GET은 URL에 값이 ?뒤에 쌍으로 이어 붙고 POST는 숨겨진 값으로 전송된다.
- GET은 URL에 이어 붙기 때문에 길이제한이 있어서 많은 양의 데이터는 보내기 어렵고 POST는 많은 양의 데이터를 보내기에도 적합하다.
- GET은 가져오는 것(Select)이고 POST는 수행(Insert, Update, Delete)하는 것입니다.

9. 쿠키(Cookie)와 세션(Session)을 정의하고 특징을 설명하시오.

- * 쿠키(cookie)
 - 클라이언트(브라우저)에 데이터를 저장, setCookie
- * 세션(Session)
 - SID(session ID)를 식별자로 서버에 데이터를 저장
 - SID로는 쿠키나 도메인 파라미터를 사용, 주로 사용자 인증시에 사용함

10. 쿠키(Cookie)와 세션(Session)의 장단점에 대하여 말해 보세요.

- 쿠키(Cookie) : 클라이언트에서 실행됨으로 보안이 취약하나 속도가 빠르다.
- 세션(Session) : 서버에서 실행되며 속도는 느리지만 보안 기능이 강력하다. 로그인 인증은 보안 때문에 세션(Session)으로 만든다.

11. "Hello Servlet"을 출력하는 ServletCode를 작성하시오..

```
public class HelloServlet extends HttpServlet {
    protected void doGet(HttpServletRequest request,
        HttpServletResponse response) throws ServletException, IOException {
        response.setContentType("text/html;charset=euc-kr");
        PrintWriter out = response.getWriter();
        out.println("<html><head><title>Hello JSP</title></head>");
        out.println("<body>Hell Servlet</body><html>");
        out.close();
    }
}
```

12. XML의 특징을 설명하시오.

XML(Extensible Markup Language)은 W3C에서 다른 특수 목적의 마크업 언어를 만드는 용도에서 권장되는 다목적 마크업 언어이다. XML은 SGML의 단순화된 부분집합이지만, 수많은 종류의 데이터를 기술하는 데 적용할 수 있다. XML은 주로 다른 시스템, 특히 인터넷에 연결된 시스템끼리 데이터를 쉽게 주고 받을 수 있게 하여 HTML의 한계를 극복할 목적으로 만들어졌다.

XML은 마크업 언어의 일종으로, 문서를 사람과 기계 모두가 읽을 수 있는 형식으로 부호화하는 규칙의 집합을 정의한다.

XML의 설계 목표는 단순성, 일반성, 인터넷을 통한 사용가능성을 강조하였다. XML은 텍스트 데이터 형식으로 유니코드를 통해 전 세계 언어를 지원한다. XML 설계가 문서에 집중하지만, 임의의 자료구조를 나타내는 데 널리 쓰이고 있다. 예를 들어 웹 서비스가 그렇다.

13. 웹 프로그래밍에서 한글 깨짐 현상이 발생하는 2가지 경우와 이를 방지하기 위해 사용할 수 있는 방법을 기술하시오.

- ★. 한글 지원 : 국내는 euc-kr과 UTF-8을 사용하고 있다
 - euc-kr : 한글과 영어 지원
 - UTF-8 : 다국어 지원
- ★. 따라서 아래에서 사용된 euc-kr과 UTF-8은 대체해도 답이 될 수 있다. 물론 같은 프로젝트 내에서는 한 가지로 통일해야 한다
 1. response(응답)
 - 1) response.setContentType("text/html;charset="UTF-8");
 2. request(요청)
 - 1) 공통 : new String(ko.getBytes("euc-kr"),"8859_1");
 - 2) post방식 request.setCharacterEncoding("UTF-8");
 - 3) get방식(Tomcat의 경우)
 - . 톰캣홈 Wconf 폴더와 이클립스의 [프로젝트 탐색기]뷰에서 [Servers]-[Tomcat v5.5 서버]항목에 있는 server.xml 파일 <Connector>태그의 속성에 URIEncoding="EUC-KR"문장 추가
 - . Server.xml Connector태그에 useBodyEncodingForURI="true"추가

14. MVC란 무엇인가?

모델-뷰-컨트롤러(Model-View-Controller, MVC)는 소프트웨어 공학에서 사용되는 아키텍처 패턴이다. 이 패턴을 성공적으로 사용하면, 사용자 인터페이스로부터 비즈니스 로직을 분리하여 애플리케이션의 시각적 요소나 그 이면에서 실행되는 비즈니스 로직을 서로 영향 없이 쉽게 고칠 수 있는 애플리케이션을 만들 수 있다. MVC에서 모델은 애플리케이션의 정보(데이터)를 나타내며, 뷰는 텍스트, 체크박스 항목 등과 같은 사용자 인터페이스 요소를 나타내고, 컨트롤러는 데이터와 비즈니스 로직 사이의 상호동작을 관리한다.

15. 아이바티스란 무엇인가? (요즘은 마이바티스로 이름을 바꿈)

iBatis(아이바티스)는 SQL에 기반한 데이터베이스와 자바, 닷넷(.NET), 루비(Ruby) 등을 연결시켜 주는 역할을 하는 영속성 프레임워크이다. 이러한 연결은 프로그램의 소스코드에서 SQL 문장을 분리하여 별도의 XML 파일로 저장하고 이 둘을 서로 연결시켜주는 방식으로 작동한다.

[웹표준 기술]

1. jQuery 란 무엇인가?

jQuery(제이쿼리)는 브라우저 호환성이 있는 HTML 속 자바스크립트 라이브러리이며 클라이언트 사이드 스크립트 언어를 단순화 할 수 있도록 설계되었다. 존 레식이 2006 년 뉴욕 시 바캠프(Barcamp NYC)에서 공식적으로 소개하였다. jQuery 는 오늘날 가장 인기 있는 자바스크립트 라이브러리 중 하나이다.

2. DOM 은 무엇인가?

문서 객체 모델(DOM; Document Object Model)은 객체 지향 모델로써 구조화된 문서를 표현하는 형식이다. DOM 은 플랫폼/언어 중립적으로 구조화된 문서를 표현하는 W3C 의 공식 표준이다.

3. Ajax 란 무엇인가?

Ajax(Asynchronous JavaScript and XML, 에이잭스)는 대화식 웹 애플리케이션의 제작을 위해 아래와 같은 조합을 이용하는 웹 개발 기법이다.

- 표현 정보를 위한 HTML (또는 XHTML) 과 CSS
- 동적인 화면 출력 및 표시 정보와의 상호작용을 위한 DOM, 자바스크립트
- 웹 서버와 비동기적으로 데이터를 교환하고 조작하기 위한 XML, XSLT, XMLHttpRequest (Ajax 애플리케이션은 XML/XSLT 대신 미리 정의된 HTML 이나 일반 텍스트, JSON, JSON-RPC 를 이용할 수 있다).

4. Node.js 란 무엇인가?

Node.js 는 확장성 있는 네트워크 어플리케이션(특히 Server-side) 개발에 사용되는 소프트웨어 플랫폼이다. Node.js 는 작성언어로 자바스크립트를 활용하며 Non-blocking I/O 와 단일 스레드 이벤트 루프를 통한 높은 처리성능을 가지고 있다.

Node.js 는 내장 HTTP 서버 라이브러리를 포함하고 있어 웹서버에서 아파치 등의 별도의 소프트웨어 없이 동작하는 것이 가능하며 이를 통해 웹서버의 동작에 있어 더 많은 통제를 가능케 한다.

5. CSS에 대하여 설명해 보세요.

- Cascading StyleSheet 의 약자로 태그 속성들을 하나의 스타일로 설정하는 용도로도 쓰인다
- HTML 문서 내에 글자의 글꼴 종류, 크기, 색, 여백 등을 지정
- 글자의 정렬 방식을 결정하거나 글자에 그림자를 지정하는 등 다양한 효과
- 사용자의 웹 브라우저 환경에 상관없이 일정한 화면을 보여줄 수 있다

6. CSS는 개발자가 다루어야 할 부분인지? 웹 디자이너가 다루어야 할 부분인지?

[Spring Framework Programming]

1. CI란 무엇인가?

continuous integration 의 약자로 말 그대로 지속적인 통합을 말한다.

개발이 완료되는 시점에 단위기능들을 통합하는 것이 아닌 개발과 동시에 통합을 진행함으로써 소프트웨어의 품질을 향상시키는 것을 목표로 한다.

2. DI란 무엇인가?

Dependency Injection 의 약자로 의존성을 주입하는 것을 뜻한다.

설정 파일을 통해 객체간의 의존관계를 설정함으로써 외부 Assembler 가 객체간의 의존 관계를 정의하게 되며, 객체는 직접 의존하고 있는 객체를 생성하거나 검색할 필요가 없어지므로 코드의 관리가 쉬워진다.

3. AOP란 무엇인가?

Aspect Oriented Programming 을 뜻하며 다양한 곳에서 자주 사용되는 공통 관심요소를 단일 기능으로 뽑아내어 코드의 중복을 줄이고 관리의 효율성을 높이는 것을 목적으로 한다.

로깅이나 로그인 등의 기능을 예로 들 수 있다.

4. POJO란 무엇인가?

Plain Old Java Object, 간단히 POJO는 말 그대로 해석을 하면 오래된 방식의 간단한 자바 오브젝트라는 말로서 J2EE등의 중량 프레임워크들을 사용하게 되면서 해당 프레임워크에 종속된 "무거운" 객체를 만들게 된 것에 반발해서 사용되게 된 용어이다. POJO라는 용어는 이후에 주로 특정 자바 모델이나 기능, 프레임워크 등을 따르지 않은 자바 오브젝트를 지칭하는 말로 사용되었다. 스프링 프레임워크는 POJO 방식의 프레임워크이다.

5. Maven(메이븐)에 대하여 말해 보세요.

- JAVA용 프로젝트 관리 툴
- 아파치(Apache) 오픈 소스 빌드 툴
- 빌드
- 문서화
- 리포팅
- 의존 관계
- 소스 코드 관리
- 릴리즈
- 배포
- 프로젝트 관리에 필요한 모든 작업을 추상화하고 표준화해서 반복을 제거
- 메이븐이 접근할 수 있는 저장소를 지원
- 이 저장소를 통해 템플릿 프로젝트인 아키타입, 의존 관계에 있는 라이브러리, 메이븐 플러그인 기능을 지원
- 프로젝트 빌드에 필요한 라이브러리, 플러그인을 저장소에서 개발자 PC로 자동으로 다운로드

6. SVN(Subversion)에 대하여 말해 보세요.

- SVN은 버전관리 시스템으로 서버기반으로 사용할 수 있는 프로그램이다.
- 로컬과 SVN서버와 커밋 넘버링 방식으로 SVN버전을 체크하여서 동기화를 하는 방식이다.

7. Hibernate에 대하여 말해 보세요.

Hibernate은 ORM(Object-Relational Mapping) 프레임워크의 한 종류이다.

많이 사용하는 iBatis도 같은 부류의 프레임워크이다.

8. O/R Mapping에 대하여 말해 보세요.

ORM 이란 객체와의 관계를 맵핑시킨다는 뜻인데, 객체와 DB 정보의 관계를 맵핑시켜 좀더 효율적으로 데이터를 처리하고자 하는 프레임워크이다.

9. iBATIS역할에 대하여 말해 보세요.

iBatis(아이바티스)는 SQL에 기반한 데이터베이스와 JAVA, 닷넷(.NET), 루비(Ruby) 등을 연결시켜 주는 역할을 하는 영속성 프레임워크이다. 이러한 연결은 프로그램의 소스코드에서 SQL 문장을 분리하여 별도의 XML 파일로 저장하고 이 둘을 서로 연결시켜주는 방식으로 작동한다.

또 다른 영속성 프레임워크인 하이버네이트(Hibernate)와 비교하여 하이버네이트는 객체모델을 사용자가 생성을 하면 프레임워크에서 데이터베이스와 연결을 시켜주는 방식인데 반해 iBatis는 사용자가 SQL 문장을 만들면 그에 적

합한 객체모델을 생성하는 방식으로 작동한다.

10. REST 서비스란 무엇인가?

확장성 생성 언어(XML) 파일로 된 웹 페이지를 읽어 원하는 정보를 수집하는 기능. 웹 페이지를 만드는 사람은 주기적으로 내용을 개정하고 사용자는 그 페이지의 URL만 알면 웹 브라우저로 읽어 정보를 얻을 수 있다. 하이퍼텍스트 전송 규약(HTTP)과 XML을 포함한 웹 기술 및 프로토콜을 사용하는 구조적 형태로서 단순 객체 접근 프로토콜(SOAP)보다 사용이 간편하고, 사이트 내용을 기술하는 RSS(RDF Site Summary)의 정보 편집 기능과 유사하다. RSS는 자원 기술 개념(RDF)을 사용한다.

11. iBatis와 MyBatis ORM 프레임워크에서 하나 이상의 레코드를 검색해서 컬렉션 리스트로 반환하는 쿼리문 실행 메소드에 대해 말해 보세요.

- iBatis : queryForList()
- MyBatis : selectList()

12. iBatis의 inset() 메소드와 MyBatis insert() 메소드의 차이점에 대해 말해 보세요.

- iBatis의 inset() 메소드는 반환값이 Object 형태 입니다.
- MyBatis insert() 메소드는 실행된 레코드 행의 수를 int 정수형으로 반환합니다.

13. iBatis의 inset() 메소드를 사용하면 반환값이 Object형이므로 저장 성공과 실패를 if문으로 판별하기가 어렵다. 그러면 iBatis에서 레코드를 저장해서 반환값을 쿼리문으로 실행된 레코드의 갯수로 반환해서 if문으로 분기 하려면 어떻게 해야 하는지 말해 보세요.

- update() 메소드를 사용하면 됩니다.

14. MyBatis를 이용할 때 쿼리의 종류에는 어떤 것이 있는지 말해 보세요.

- 한건 조회 : selectOne
- 여러건 조회 : selectList

[Project]

1. SI가 무엇인지 말해 보세요.

SI(system integration) = 시스템통합

기업의 경영목표 달성을 위해 정보시스템을 구축하는 종합서비스를 말한다. 하드웨어, 소프트웨어, 통신망, 전산인력 등의 전산자원을 일의 목적과 특성에 맞게 통합해 최적의 해결점을 제시하고 정보시스템을 개발·유지·보수하는 과정까지 포함하는 광범위한 개념이다.

기업이 필요로 하는 정보시스템에 관한 기획에서부터 개발과 구축, 나아가서는 운영까지의 모든 서비스를 제공하는 일.

2. 개발 프로세서에 대하여 말해 보세요.

- ISP(information strategic planning) : 전략정보시스템 계획)
- BPR(business process re-engineering) : 업무 재설계
- 요구사항분석
- 업무분석
- 설계
- 구현(coding) ; 단위시험 포함
- 시험 : 통합시험, 시스템 시험, 인수 시험
- 인수 및 구축(운영서버에 시스템 구축 및 데이터 이행)
- 운영 및 유지 보수

3. 프로젝트에 대하여 설명하고 본인의 역할은 무엇이었는지? 불화는 없었는지?

목적, 기대효과와 주요 메뉴를 설명하고 자신의 업무 설명
의견이 대립 될 때 슬기롭게 타협한 과정 설명

4. 이 작품에서 남들이 쉽게 따라 할 수 없는 자신만의 노하우(기술)는?

만족할 만한 부분소개

5. 이 작품에서 좀더 보강되어야 할 부분이 있는지?

시간이나 기술력 부족으로 완성하지 못했으나 추가하고 싶었던 부분

6. Spring을 사용한 이유에 대해 말해 보세요.

Spring은 xml이나 annotation등을 이용한 선언적 관리가 가능하다.

개발의 복잡성이 감소했고, 특정환경에 제약을 받지 않으며, 테스트와 관리 및 유지 보수가 좋아지기 때문에

7. MVC2 모델을 사용한 이유에 대해 말해 보세요.

설계는 다소 시간이 필요하지만 프로그램 개발자와 디자이너가 전달되는 데이터만 선언되어 있다면 서로 상관없이 각자 따로 동시에 개발이 가능하다.

8. Oracle(MySQL 등)을 사용한 이유에 대해 말해 보세요.

Oracle

- JAVA 개발 시 호환성과 안정성을 중점으로 두고 있으며 그기에 적합한 Oracle을 주로 많이 사용하므로

MySQL

- 저렴한 가격으로 중소형 Project에서 많이 사용하고 다루기 쉬운 옵션을 제공한다.

MS-SQL

- windows 서버를 사용하는 경우 적합한 데이터베이스로 일반관리 사용자가 익숙한 windows와 유사하므로 데이터베이스관리자가 선호할 수 있다.

9. 가장 힘들었던 경험과 그것을 어떻게 극복했는지?

10. JSP 외에 다른 프로그래밍 언어를 공부해 본 경험이 있는지?

11. 최근에 관심을 가지고 공부하고 있는 분야는?

[빅데이터, 인공지능]

1. 어떤 데이터를 다루어 보셨나요?

- 텍스트 데이터
- 이미지 데이터
- 숫자 데이터
- 음성 데이터

2. 데이터를 수집, 정제, 변환 해 본 경험이 있나요?

- 크롤링(Crawling), Flume, Spark

3. 빅데이터 분석 및 시각화를 해 본 경험이 있나요?

- R / R Studio
- Python

4. 딥러닝 프레임워크는 어떤 것을 사용해 보셨나요?

- TensorFlow
- Keras
- PyTorch(Torch)
- Caffe
- Deep Learning 4j- Theano
- MxNet
- CNTK
- Lasagne
- Big DL

5. 인공지능 개발도구는 어떤 것을 사용해 보셨나요?

- 파이썬, R, JAVA, C/C++, Javascript, Lua, Julia, Swift

6. TensorFlow 2.0을 사용해 보셨나요? 어떤 부분이 이전과 달라졌나요?

- Keras와 즉시 실행(eager execution)을 이용한 쉬운 모델 작성
- 어떤 플랫폼에서든 튼튼한(robust) 모델 배포
- 연구를 위한 강력한 실험
- deprecated된 API를 정리하고 중복을 줄임으로써 API 단순화

7. 다루어 본 인공지능 알고리즘에는 어떤 것들이 있나요?

머신러닝 알고리즘

- 지도 학습(Supervised learning)
- 준지도 학습(Semi-supervised learning)
- 비지도(자율) 학습(Unsupervised learning)
- 강화 학습(Reinforcement learning)

딥러닝 알고리즘

- 심층 신경망(DNN, Deep Neural Network)
- 합성곱 신경망(CNN, Convolutional Neural Network)
- 순환 신경망(RNN, Recurrent Neural Network)
- 제한 볼츠만 머신(RBM, Restricted Boltzmann Machine)
- 심층 신뢰 신경망(DBN, Deep Belief Network)

8. 파이썬 언어로 데이터 모델링을 해 본적이 있나요?

9. 리스트 딕셔너리 튜플 차이

10. append, extend의 차이

11. 퍼셉트론에 대한 기본 정의
12. 딥러닝 중 XOR 게이트의 한계, 극복 방안
13. 딥러닝 알고리즘 종류
14. 비지도, 지도 학습의 차이
15. 아웃풋의 범주형 자료를 숫자형으로 바꾸기 위한 방법 - 원핫인코딩
16. 과적합이 생기는 원인, 방법
17. 협업필터링 - 유사도 결정 시 유클리드 거리와 코사인 유사도의 차이. 어떤 때 어느 것을 써야하는지
18. 협업필터링은 지도학습? 비지도학습? 이유는?
19. 추천시스템 알고리즘의 종류와 대략적 설명
20. 추천시스템은 누구나 하는데, 자기만의 전략은 무엇인지?