**기술 면접**

**자바)**

1. **가비지 컬렉션이란**
2. **String과 StringBuffer의 차이**
3. **오버로딩과 오버라이딩의 차이**
4. **추상클래스와 인터페이스 차이**
5. **제네릭이란?**
6. **예외처리의 목적,정의,종류**
7. **Wrapper Class**
8. **기타 (소프트웨어 패턴) : 싱글턴 패턴 / 팩토리 패턴**
9. **DAO와 Service의 차이**

**오라클)**

1. **스키마(Schema)**
2. **뷰**
3. **인덱스 생성시 고려할 사항**
4. **테이블을 드롭(DROP)하는 것과 자르는 것(Truncate), 그리고 테이블 내 모든 레코드를 삭제(Delete)하는 것의 차이점**
5. **DB에서의 Commit와 Rollback 이란?**
6. **DB정규화란(개요),정규화(목적)**
7. **RDB에서 Join이 무엇인지에 대한 설명과 간단한 샘플 쿼리문을 작성해보아라**
8. **DB Transaction(트랜잭션) 이란?**

**JSP)**

1. **Servlet의 동작 과정 설명**
2. **GET / POST의 차이점**
3. **쿠키와 세션의 차이**
4. **HTTP Method 4가지설명**
5. **Rest API에대해설명하시오**
6. **XML과 JSON,JSONP의 장단점**
7. **UTF-8 , EUC-KR의 차이점**
8. **Data-속성은 무엇이고 언제 사용하는가(자바스크립트)**
9. **Class와 id의 차이점(자바스크립트)**
10. **AJAX란 (자바스크립트)**
11. **Jquery를 한마디로 정의한다면 (자바스크립트)**
12. **동기화와 비동기화란 (자바스크립트)**
13. **Prototype,크로저에 대해 설명 (자바스크립트)**
14. **MVC의 동작 구조와 MVC를 사용했을 때 장단점**

**Spring)**

1. **Framework란 (장점,단점)**
2. **SpringFramework**
3. **Spring 에서 어노테이션사용과 XML사용의 차이점**
4. **Spring MVC의 동작 과정**
5. **Spring DI**
6. **Spring AOP**
7. **RESTFUL이란?**
8. **Spring Transaction 처리 방법**
9. **Spring-Security 처리 과정**
10. **Spring VS Spring Bootml 차이점**

**Front)**

1. **Ajax란**
2. **VueJs의 장단점 , 사용처**
3. **ReactJs의 장단점 , 사용처**
4. **클로저란?**
5. **호이스팅**
6. **React-Query의 장단점**
7. **ES6에서 추가된 내용**

**프로젝트 관련)**

**협업**

**GIT**

**담당파트 부분에 대한 설명**

다음 중 자바스크립트의 특징으로 옳지 않은 것은?

A. 인터프리터 언어이다

B. 동적 타이핑을 지원한다

C. 서버 전용 언어이다

D. 비동기 처리를 지원한다

2. HTTP 상태코드 404는 무엇을 의미하는가?

A. 서버 오류

B. 권한 없음

C. 요청한 자원을 찾을 수 없음

D. 성공

1. REST API의 주요 구성 요소 3가지를 설명하세요.

2. 쿠키(Cookie)와 세션(Session)의 차이를 설명하시오.

[문제] 숫자 배열이 주어졌을 때, 중복을 제거하고 오름차순으로 정렬하시오.

입력: [3, 1, 2, 3, 4, 2]

출력: [1, 2, 3, 4]

function deduplicateAndSort(arr) {

return [...new Set(arr)].sort((a, b) => a - b);

}

[문제] EMPLOYEE 테이블에서 부서ID가 10인 직원들의 이름(name)과 급여(salary)를 조회하는 SQL을 작성하시오.

SELECT name, salary FROM EMPLOYEE WHERE department\_id = 10;

1. TCP와 UDP의 차이를 설명하시오.

2. 스택과 큐의 차이를 예시와 함께 설명하시오.

Node.js, React, Java Spring