Q. 객체지향설계원칙 중 단일책임원칙에 대해서 설명해주세요

Q. 빅테크 기업의 의미를 설명해 주세요.

금융 관련 언론 및 국제기구에서 금융서비스 또는 금융상품과 매우 유사한. 상품을 직접 제공하는 거대 기술회사를 지칭하는 용어

Q. 알고 있는 개발자의 종류를 적어주세요.

Q. Java의 연산자 "=="와 "equals()"의 차이점을 설명해 주세요.

Q. Java의 실수형 데이터 타입을 적어 주세요.

Q. Java와 Spring의 관계를 설명해 주세요.

Q. IDE에 대해서 설명해 주세요.

Q. 테스트주도개발(TDD)에 대해서 설명해 주세요.

J-unit을 먼저 개발하고 코드를 개발해야 정확한 개발

Q. 애자일(Agile) 방법론에 대해서 설명해 주세요.

Q. Frond-End와 Back-End의 차이점을 설명해 주세요.

Q. 스프링의 특징, 3대 장점

Q. 객체지향 설계원칙(솔리드(SOLID) 5원칙)

Q. MVC

각 계층별 영향도를 최소화하기 위함

Q. DDL/DML/DCL 이란?

Q. SQL에서 인덱스(Index)

pk가 기본적으로 인덱스로 설정

그러나 별도의 인덱스를 만들 수는 있다.

(인덱스 생성시 인덱스 기준의 가상의 테이블생성)

Q.스키마

Q.깃에서 커밋이란?

Q.애자일, 폭포수 개발방법

Q. ITO (Information Technology Outsourcing)

전산업무나 정보자원의 일부 또는 전부를 외부의 전문기업에 위탁 운영하는 것

Q. SM (System Maintenance) / AMS

시스템 유지보수

Q. SI (System Integration)

시스템 통합 및 신규구축

Q. IT거버넌스 (IT Governance)

기업의 전략과 목표에 부합되도록 IT와 관련된 Resource와 Process를 통제/관리하는 체계

IT활용에 있어 바람직한 행동을 지원하기 위한 의사결정 및 책임에 대한 Framework

Q. ITSM(서비스관리시스템, IT Service Management)

Q. SVN(형상관리시스템) subVersion

여러명이서 작업하는 프로젝트의 버전관리나 각자 만든 소스의 통합과 같은 문제를 해결하기 위해 저장소를 만들어 그곳에 소스를 저장해 소스 중복이나 여러 문제를 해결하기 위한 형상관리/소스 관리 툴이다.

Q. CI/CD (Continuous Integration/Continuous Delivery)

지속적인 통합, 지속적인 서비스 제공, 지속적인 배포

Q. DevOps (개발+운영 통합 관리)

애플리케이션과 서비스를 빠른 속도로 제공할 수 있도록 조직의 역량을 향상시키는 문화 철학, 방식 및 도구의 조합

개발해서 소스코드를 형상관리에 올리면 운영서버, 개발서버, 테스트 서버까지 반영되어 피드백까지 빠르게 이행됨

구조를 긴밀, 간결, 실시간으로 만든것

Q. MSA (MicroService Architecture) 마이크로서비스 아키텍처

작고, 독립적으로 배포 가능한 각각의 기능을 수행하는 서비스로 구성된 프레임워크라고 할 수 있다.

마이크로서비스는 완전히 독립적으로 배포가 가능하고,

다른 기술 스택(개발 언어, 데이터베이스 등)이 사용 가능한 단일 사업 영역에 초점을 둔다.

애플리케이션 개발을 위한 아키텍처 스타일을 의미합니다.

마이크로서비스를 사용하면 대규모 애플리케이션을 각각 담당 영역을 가진 소규모의 독립적인 구성요소로 구분할 수 있습니다.

Q. Jenkins

지속적 통합 시스템(빌드/테스트/배포 자동화)

Q.SonarQube(BXQ#)

품질 모니터링, 정적 분석(코드가 클린한지 자동 체크)

Q. ITSM (IT Service Management)

서비스들을 계획, 설계, 전달, 운영하기 위해, 단체에 의해 수행되는 활동 전반

(정책에 의해 감독, 프로세스를 통해 조직 및 구성, 절차 지원)

Service들의 관리 체계

Q. ITIL (IT Infrastructure Library)

IT서비스 관리에 대한 Best Practice 집합

사실상의 산업표준

Q. ISO 20000 (ISO :국제 표준화 기구)

ITIL기반, IT운영체계를 평가하는 ITSM 국제 심사 인증 표준

Q. 서버(Server)/클라이언트(Client, 로컬)

C/S 프로그래밍 : 서버와 클라이언트 구조로 작동하는 프로그램

서버가 중앙에 있고 클라이언트 여러대가 붙어서 작동

2 Tier

서버 - 클라이언트

Q. 3 tier architecture

서버 - WAS -( 웹서버 )- 웹브라우저

Web과 WAS(Web Application Server)의 차이점

웹서버와 WAS가 출연하면서 미들웨어가 출연

ex. 톰캣 등등

Q. UML( Unified Modeling Language )

모델을 만드는 표준언어

ex. 컨텍스트 다이어그램, 클래스 다이어그램, 시퀀스 다이어그램... 등등

Q. ERP 시스템(Enterprise Resource Planning)

ERP(전사적 자원 관리)는 조직이 최적의 성능을 위해 핵심 비즈니스 프로세스를 자동화하고 관리할 수 있는 소프트웨어 시스템입니다. ERP 소프트웨어는 기업의 비즈니스 프로세스 간의 데이터 흐름을 조정하여 단일 데이터 소스를 제공하고 엔터프라이즈 전반의 운영을 간소화합니다.

Q. 3 M/M (Man per Month) = 공수:Effort

M/W (Week)

M/D (Day)

Q. WBS(Work Breakdowm Structure) : 업무 분업 구조, 작업 분해 구조

프로젝트 시작할 때 작성

Q. network area

(규모에 따라 나눔)LAN, WAN, MAN

VAN(value added network) 부가가치통신망 ex. 은행공동망

Q. 테스트 설계 기법에 따른 테스트의 종류(방식)

\* White Box Test = 구조 기반 테스트케이스 설계

코드와 개발 설계 등의 소프트웨어 구현 정보를 기반으로 테스트케이스를 도출한다. 잘 보임, 빠짐없이 설계 가능

수행된 테스트케이스를 바탕으로 테스트커버리지(Test Coverage, 얼마나 꼼꼼하게 했나)를 측정할 수 있다

테스트 커버리지 7~80% -> 잘했다

\* Black Box Test = 명세 기반 테스트케이스 설계

명세(Specification)를 기반으로 테스트케이스를 도출한다.

커버리지를 측정할 수 있으나, 그 의미가 구조 기반 기법의 커버리지에 비해 제한적이다.

Q. trouble shooting / debugging

문제가 생겼을 때 해결해나가는 과정

에러를 잡는 과정

Q. IoC, injection 등등 스프링의 특징

Q. MCI(Multi Channel Integration)

여러 채널을 통합하는 미들웨어 솔루션

Q. 채널

사용자가 서비스를 접근하는 방식

Q. DBMS

Q. EAI (Enterprise Application Integration)

은행 안 시스템들 중 하나가 코어뱅킹

여러 시스템들이 통신하도록 해주는 방식,기술

Q. ESB (Enterprise Service Bus)

EAI보다는 작은 단위

그 프로그램 안의 특정 기능과 기능 간의 통신

Q. FEP (Multi Channel Architecture)

대외솔루션

내부시스템과 외부 대외기관의 시스템을 연계해주는 대외계 솔루션

Q. BRE

Q. BPM (Business Process Management)

Q. JVM (Java Virtual Machine)

Q. DTO = OMM (I/O 정의)

Q. 핫디플로이(Hot Deploy배포)기술

서버를 재기동하지 않고도 프로그램을 안정적으로 적용할 수 있게 하는 기술

Plug&Play(꼽으면 작동한다)

Q. BXI(통신을 도와주는 제품)

Q. APM(Application Performance Management)

어플리케이션 성능 관리

Q. 프로토콜(통신규약)

http(Hyper Test Transfer Protocal)

ftp(File Transfer Protocol)

Q. URL(Uniform Resource Locator)

Q. get, post

Q. transaction/traffic

: request <-> reponse

Q. 배치(신입 때는 할일없음)

tasklee

chunked

Q. Code Inspection(검사)

개발팀에서 작성한 개발소스 코드를 분석하여 개발 표준에 위배되었거나 잘못 작성된 부분을 수정하는 작업을 말한다.

위배된 코딩(ex 3중포문)을 제약하는 기능

Q. 정적 분석

ex. warning

Q. postman

API 방식으로 개발된 웹서비스를 호출해주는 프로그램

request - response 웹 실행 테스트

Q. web application

Q. REST API

자바 웹 프로그램을 개발하는 방법 중 하나

Q. COMMIT

Q. ROLLBACK

Q. Request

Q. Response

Q. Rest API 방식의 핵심:

'stateless' 상태값을 가지고 있지 않기 때문에 파라미터에 따라 리턴을 한다

Request를 뭐로 주느냐에 따라서 Response가 결정됨

Session이나 Cookie를 사용하지 않음

Q. GIGO

Garbage In Garbage Out 쓰레기가 들어가면 쓰레기가 나옴

Q. POJO기법

POJO Class(getter, setter만 있는 클래스)

POJO(Plain Old Java Object) : Java로 생성하는 순수한 객체

VO = DTO(Data Transfer Object)

둘 다 data transfer용도

VO는 DB I/O가 있을 때(db에 접근할때) 사용

business logic이 없어야 함

Q. EJB(Enterprise Java Bean)프레임워크

EJB는 자바 기반의 엔터프라이즈 애플리케이션 개발을 위한 서버 측 컴포넌트 모델입니다.

이를 통해 큰 규모의 비즈니스 애플리케이션을 구축하고 관리할 수 있습니다.

Q. CBD 방법론

CBD 개발방법론(Component Based Development)이란 컴포넌트를 조합해 재사용함으로써

개발 생산성과 품질을 높이고 시스템 유지보수 비용을 최소화할 수 있는 개발방법론이다.

컴포넌트 기반 개발방법론이라고 한다.

Q. J2EE

J2EE란? J2EE는 자바 기술로 기업환경의 어플리케이션을 만드는데 필요한 스펙들을 모아둔 스펙 집합입니다.

Q. JDK(Java Development Kit)란? 자바로 개발하기 위한 라이브러리 집합

JDK는 자바 개발키트(Java Development Kit)의 약자로 이름 그대로 개발자들이 자바로 개발하는 데 사용됩니다.

JDK안에는 개발 시 필요한 라이브러리들과 javac, javadoc 등의 개발 도구들을 포함되어 있고

개발을 하려면 당연히 실행도 시켜줘야 하기 때문에 JRE (Java Runtime Environment)도 함께 포함되어 있습니다.

Q. JRE(Java Runtime Environment)란? 자바로 개발된 프로그램 실행 라이브러리

JRE는 자바 실행 환경(Java Runtime Environment)의 약자로

자바로 만들어진 프로그램을 실행시키는데 필요한 라이브러리들과 각종 API, 그리고 자바 가상 머신 (JVM)이 포함되어 있습니다.

JRE는 자바로 "개발(쓰기)은 안되고 실행(읽기)만 된다"라고 생각해주시면 될 것 같습니다.

Q. 리팩토링(Refactoring)

Q. enum class

가독성 목적

Q. API

Q. conflict mode

Q. merge

Q. 선점형

Q. 비선점형

Q. 배포 deployment

Q. Agile manifesto

Q. 좋은 프로그램

원래 목적한 기능을 수행하는 프로그램

설계(Architecture, Design)가 우수한 프로그램

성능(Performance)이 뛰어난 프로그램

결함(Defect), 오류(Error)가 없는 프로그램

결함 - 요구사항을 만족시키지 못하는 것

가독성(Readability)이 좋은 프로그램

주석이 잘 정리된 프로그램

유지보수가 용이한 프로그램

Q. 프로그램 분석기법

정적 기법 - ex. PMD 코드분석 자동화 프로그램 등

서버에 반영되기 전에 확인 가능

동적 기법 - ex. enpharos, APM(어플리케이션 성능 모니터링) 등

but, 이미 서버에 반영된 뒤에 확인 가능

Q. 클래스 로딩 순서

Q. pull up method -> 메소드로 빼서 상위클래스로 옮겨라

하위클래스에서 상속받아서 사용

Q. Inline class

클래스가 하는 일이 많지 않은 경우에는

그 클래스에 있는 모든 변수와 메소드를 다른 클래스로 옮기고 그 클래스를 제거하라

Q. 기본타입, 참조타입

Q. 디자인패턴

Q. 싱글톤

Q. 객체지향과 관점지향 프로그래밍