

Chapter 04. 프로그래밍 언어별/각 IT 분야별 문제 유형

프로그래밍 언어별/각 IT 분야별 문제 유형

목표: 신입 기술면접 준비 방법 익히기

- 기술 면접의 유형과, 문항 유형에 대해 정리하고,
- 각 문항 유형별 예시와 답변 예시를 알아보고,
- 답변 가이드 라인과 팁을 정리하며,
- 각 기술면접 유형과, 문항 유형 준비 방안에 대해 소개

문제 유형

1. 문제 유형1 - 자기 소개 관련
2. 문제 유형2 - CS 기본 지식 관련
3. **문제 유형3 - 프로그래밍 언어별/각 IT 분야별 세부 지식 관련**
4. 문제 유형4 - 자료구조/알고리즘 기반 문제 손코딩

각 유형별 주요 문제/예시 답변/팁 알아보기

- 왜 그런 문제를 물어보나?
- 어떤 답변을 기대하는가?
 - 예시 질문 및 답변
- 팁은 무엇인가?

주요 질문1 (JAVA)

1. JVM과 JAVA 프로그램 실행 과정을 설명해보세요

JAVA 프로그램은 JVM 상에서, 실행되므로 JVM과 JAVA 프로그램의 구조/실행 방식을 잘 이해하고 있어야 보다 효율적인 프로그램 작성이 가능하므로, 이와 관련된 지식을 판단하고자 함

주요 답변

- JVM이란 JAVA Virtual Machine(자바 가상 머신)의 약자로, 자바 프로그램을 자바 API를 기반으로 실행하는 역할을 함
- JAVA 프로그램 실행 과정은
 - 프로그램이 실행되면 JVM이 OS로부터 해당 프로그램이 필요로 하는 메모리를 할당받고,
 - 자바 바이트코드로 변환된(.class) 파일을 **class** 로더를 통해 **JVM**에 로딩한다.
 - 로딩된 **class** 파일은 **execution engine**을 통해 해석되고, 실행된다.
 - 필요시 garbage collection을 수행해서, 불필요하게 할당된 메모리를 해제한다.

주요 질문2 (JAVA)

2. Garbage Collection 이 필요한 이유

실무에서 JAVA 프로그램을 작성할 경우, 간혹 GC 문제로 인해 프로그램이 정상적으로 실행되지 않는 경우가 있음. 이런 실무적인 어려움으로 인해, GC에 대해 기본적인 내용을 잘 알고 있는 사람은 JAVA 프로그램 작성에 경험이 어느 정도 수준에 올라와 있다고 생각하므로, 이를 판단하고자 함

주요 답변

- java 프로그램은 메모리를 명시적으로 지정해서 해제하지 않기 때문에, Garbage Collection Mechanism을 통해, 경우에 따라 더 이상 필요없는 객체를 찾아 지우는 작업을 수행한다.

참고 (Garbage Collection 동작 방식)

- JVM 메모리 영역
 - JVM은 운영체제로부터 할당받은 메모리 영역을 세 영역으로 분리함
 - 메소드 영역, JVM 스택, 힙 영역
 - 이 중에서 힙 영역에 생성된 객체가 저장되며, 사용하지 않는 객체를 GC를 통해 삭제함
 - JVM 힙 영역은 다음과 같이 나뉨
 - YOUNG, OLD, Permanent Generation
 - YOUNG **generation**: eden, S0, S1 (Survivor space)
- Garbage Collection 동작 방식
 - 새롭게 생성된 객체는 YOUNG의 eden 영역에 들어가고, eden 영역이 다 차면 minor GC가 발생
 - GC가 실행되면, GC를 실행하는 스레드 외에 나머지 스레드는 멈춘다.
 - 불필요한 객체는 삭제되고 아직 필요한 객체는 S0으로 이동, S0에 있었던 객체는 S1로 이동, S1이 다 차면 S1에 아직 필요한 객체는 OLD generation으로 이동
 - OLD generation은 크기가 크므로, 이 영역이 다 차는 경우는 자주 발생하지 않음.
이 영역을 삭제할 때 major GC(혹은 full GC) 발생
 - minor GC는 자주 발생하지만, YOUNG 영역은 OLD 영역보다 적기 때문에, 프로그램 중지 시간(stop-the-world)은 짧아짐
 - YOUNG 영역을 다 비우므로, YOUNG 영역에는 연속된 여유 공간이 만들어짐

주요 질문3 (JAVA)

3. Overriding VS Overloading

언어 용어 관련 문제로, 실무에 필요한 질문이라기 보다는, 해당 언어를 사용해봤는지, 가볍게 물어보기 쉬워서 물어보는 질문

주요 답변

- **Overriding:** 상위 클래스에 존재하는 메서드를 하위 클래스에 맞게 재정의하는 것 (메서드 이름 및 파라미터수 동일)
- **Overloading:** 두 메서드가 같은 이름을 가지고 있으나, 파라미터 수나 자료형이 다른 경우

주요 질문4 (JAVA)

4. interface 와 abstract

보다 복잡한 JAVA 프로그램을 작성하다보면, **interface** 와 **abstract**를 사용해봤을 것이고, 따라서 이에 대해 어느 정도 이해하고 있는지를 기반으로 자바 프로그래밍 경험치를 판단하기 위한 질문

주요 답변

- **abstract**: 추상 클래스는 추상 메소드를 1개 이상 가지고 있는 클래스를 의미.
기존 메소드 이외에 추상 메서드를 상속시켜서, 반드시 구현이 필요한 내용인 추상 메서드를 상속받은 클래스에서 구현시키는 것이 주목적.
추상 메소드를 1개 이상 가진 클래스는 객체 생성이 안되므로,
추상 클래스를 상속받은 클래스의 객체 생성을 위해서는 추상 메서드를 구현해야 함
(추상 메서드 구현을 강제함)
- **interface**: 상수와 메소드 선언 집합, 정의된 메소드를 implements 받은 곳에서 모두 구현을 강제함
JAVA는 단일 상속만을 지원하기 때문에, 추상 클래스를 상속받은 클래스는 다른 클래스를 상속받을 수 없음
그러나 인터페이스는 인터페이스끼리 상속이 가능하므로, 여러 인터페이스 상속이 가능함

주요 질문5 (JAVA)

5. 디자인 패턴이 무엇인지, 그리고 싱글톤에 대해 간단히 설명하세요

디자인 패턴을 선호하는 개발자의 경우, 디자인 패턴에 대해 잘 알고 있는지를 판단하기 위해 질문하는 것임 (개발자마다 선호하는 개발 스타일이 다르고 이에 따라, 질문이 달라질 수 있음)

주요 답변

- 디자인 패턴은 공통적인 소프트웨어 코드 작성 문제를 해결하는데 도움이 될 수 있는 코드 패턴을 의미함
- 싱글톤은 전체 프로그램에서 단 1개의 객체를 생성해서 공유할 수 있는 코드 패턴

주요 질문6 (python)

6. 파이썬 generator에 대해 설명해주세요

파이썬은 초보자들도 이해하기 쉽지만, 세부 기능에 들어가면 이해하기 어려운 고급 기능들도 많은 언어임. 이 중에서 **generator**도 이해하기 쉽지 않은 기능 중 하나로 언급되기 때문에 질문을 통해 파이썬의 이해 정도를 알고자 하는 것임

주요 답변

- Generator는 **Iterator**를 생성해주는 함수로, 함수 안에 **yield** 키워드를 사용함.
Iterator는 `next()` 메소드를 이용해 데이터를 순차적으로 접근할 수 있는 함수임
- Generator는 한번에 모든 데이터를 메모리에 적재할 필요가 없어서 메모리 효율이 높고, 계산 결과가 필요할 때까지 계산을 늦출 수 있으므로, 수행 시간이 긴 연산을 필요한 순간까지 늦출 수 있다는 장점을 가지고 있음

주요 질문7 (python)

7. 파이썬의 GIL 에 대해 설명해주세요

파이썬의 한계점, 그리고 그 이면의 CS 지식, 그리고 파이썬을 일정 수준 이상 깊이 있게 사용했음을
검증하기 위한 질문

주요 답변

- GIL은 한번에 하나의 스레드만 수행할 수 있도록 인터프리터에 **lock**을 거는 기능
- 파이썬 객체는 Garbage Collection 기능을 위해, **reference count**를 가지고 있는데, 해당 객체를 참조할때마다 **reference count** 값을 변경해야 함
- 멀티 스레드를 실행하게 되면, 각 스레드가 공유하는 객체들에 대해 각각 **lock**을 거는 경우, 성능상 이슈와 deadlock 과 같은 위험이 존재하여, 인터프리터 레벨에서 한 시점에 실행하는 스레드는 1개로 제한한 것임.
그래서 멀티 CPU 환경이라 할지라도, 파이썬 스레드는 어느 시점에나 1개의 스레드가 실행되는 단점이 있음
- 이를 해결하려면 multiprocessing 라이브러리를 사용하면 됨.
이 경우 개별 프로세스가 생성되고, 프로세스 별로 인터프리터 락이 걸리기 때문에, 동시 실행이 가능함

주요 후보 문항: JAVA

1. 객체와 클래스의 차이점에 대해 설명해주세요
2. JAVA 메모리 영역에 대해 설명해주세요
3. `private`, `protected`, `public`, `default` 키워드에 대해 설명해주세요
4. 객체지향 5대 원칙에 대해 설명해주세요

파이썬은 기술 면접을 통해 내부적인 기술적 내용에 대해 묻기보다는 손코딩등에서 활용되는 경우가 많음

주요 후보 문항: 웹 (프론트엔드 + javascript)

1. MVC 패턴에 대해 설명해주세요.
2. 브라우저의 렌더링 과정에 대해 설명해주세요
3. 가상 DOM과 DOM의 차이점에 대해 설명해주세요
4. ES6에 추가된 스펙에 대해 생각나는 것 몇 가지만, 설명해주세요
5. var와 let/const의 차이점에 대해 설명해주세요

웹은 굉장히 실무적인 면접이 많음. 주로 프레임워크의 개별 기능을 써봤는지, 개념과 장단점에 대해 묻는 경우가 많음

주요 후보 문항: 운영체제 & 데이터베이스

1. deadlock의 개념 및 deadlock 해결 방법에 대해 설명해주세요
2. 뮤텍스와 세마포어의 차이점에 대해 설명해주세요
3. 가상 메모리에 대해 설명해주세요
4. 컨텍스트 스위칭에 대해 설명해주세요
5. Transaction의 개념에 대해 설명해주세요
6. 관계형 데이터베이스 시스템과 NoSQL의 차이점에 대해 설명해주세요

주요 후보 문항:네트워크

1. OSI 7계층에 대해 설명해주세요
2. HTTP와 HTTPS의 차이점에 대해 설명해주세요
3. 쿠키와 세션의 차이점에 대해 설명해주세요
4. RESTful의 개념에 대해 설명해주세요
5. TCP의 3-way-handshake와 4-way-handshake 방식의 차이점에 대해 설명해주세요

주요 후보 문항:자료구조 & 알고리즘

1. 배열과 링크드 리스트의 장단점에 대해 간략히 설명해주세요
2. BST의 최악의 시간 복잡도와 최악의 시간이 걸리는 케이스에 대해 설명해주세요
3. 해쉬 테이블에 대해 설명해주세요
4. Fibonacci 공식을 **recursive**와 **dynamic programming**으로 구현시 차이점에 대해 설명해주세요.
5. DFS와 BFS에 대해 간략히 설명해주세요

자료구조와 알고리즘은 코딩 테스트를 통해 검증하는 경우가 더 많음