**(일반적인, 회사질문)**

1. **우리 회사에 대해 아는가?**

마이그레이션, 통합운영, 빅데이터, 기술지원, 인공지능

DC Comics 과학 연구소 이름 따온 것.

클라우드 컨설팅 전문기업.

서비스 플랫폼 – aws, Alibaba cloud, google cloud, microsoft azure

퍼블릭 클라우드 관련 개발 및 운영 프로젝트 수행

엔터프라이즈급 고객사 파트너.

1. **집이랑 거리가 먼데 괜찮나?**

잠실 -> 역삼역(6번출구)

1시간 정도 충분. 학원이 1시간 30분거리 -> 출석률99%

1. **언어 새로 배워도 괜찮나?**

새로운 경험과 자기발전에 있어 낯선 것을 접하는 것이 어렵지 않고 즐겨한다. 새로 언어를 배워도 잘해낼 자신있다.

1. **수색대는 왜 지원했는가?**

새로운 경험, 남들이 해보지 못한 경험 같은 것들에 많은 의미를 부여하는 성격이다. DMZ라는 대한민국 1%만이 경험 할 수 있는 기회를 놓치고 싶지 않았고, 도전적인 성격이라 위험의 요소가 있는데도 불구하고 극복하는 과정에서 성장 할 수 있는 계기가 될 것 같아 지원했다.

1. **간단한 자기소개 및 자신의 성격 장단점**

반갑습니다. 이번에 스타랩스에 면접을 보게된 24살 동준영이라고 합니다.

저는 자바를 통한 Spring게시판 만들기, 그리고 Android를 통해 몇몇 프로젝트를 진행해본 경험이 있습니다.

그중 가장 기억에 남는 것은 기존에 운영하고 있는 사이트를 크롤링하여 해당 사이트 기능구현 및 관리자 페이지 운영이였습니다.

제가 맡은 역할은 관리자페이지 구현이였는데요, 어떠한 요청에 따른 결과, 사용자가 이용할 수 있는 서비스를 직접 만들고 관리자로서 관리도 하며, 어떤 웹사이트 상의 구조를 알게 된것에 많은 매력을 느꼈습니다.

또한 자바언어를 통한 안드로이드 어플 제작을 통해 실상에서 가장 많이 사용하는 어플을 직접 만들고 서비스를 제공할 수 있다는 것에 흥미를 느꼈습니다.

그리고 학원 수료가 6월 말쯤이였고, 그 뒤로 자바에 대한 기본적인 부분을 더욱 공부하고 싶어서 github를 통해 1일1커밋은 아니여도, 일주일에 6커밋하며 공부하고 있었고, 컴퓨터 공학에 대한 전반적인 지식을 얻고 싶어서 학점은행제를 통한 학사취득 준비중에 있습니다.

저는 저를 돌아보는 능력이 뛰어나다고 생각하기에, 부족한 부분을 스스로 채우려고 하고, 발전하려고 노력하는 사람이라고 생각합니다. 이 능력을 이 스타랩스에 기여하고 싶습니다. 감사합니다.

1. **당신을 뽑아야 하는 이유는?**

**회사의 인재상이 Positive, Team work, Diligent**

회사의 인재상 Positive, Team work, Diligent이 세가지에 적합한 사람이기 때문입니다.

주위에서 낙천적이라는 소리도 들어봤는데요 ㅋㅋ, 무언가에 도전하거나 어떤 어려운걸 하고있거나, 주위에 힘이 필요한 사람이 있을 때 이 긍정적인 사고가 정말 많이 도움된 것 같습니다. 안되는 것조차 될수 있도록 힘을 주는 긍정적인 태도의 가치를 정말 높이 생각하고 매사에 긍정적인 태도를 겸하려고 합니다.

또한 한때 스포츠 인으로서 팀워크의 가치를 잘 알고 있습니다. 물론 스포츠와 업무에서의 협업은 다르지만, 같이의 가치는 같다고 생각합니다.(팀원에 대한 믿음, 팀원에게 의지도 할 수 있고, 도움도 줄 수 있는 그런 시너지)

마지막으로 성실함. 사실 이 성실함을 보여드리기 위해선 직접 사람을 겪어봐야 하지만, 전에 다녔던 학원의 출석률이 99%이기도 했고, 현재 보여드릴 거라곤 학원이 끝나고 7월 15일부터 github를 하며 공부해온 흔적?밖에 없을 것 같습니다. 추가적으로 운동하던 시절과 3수시절에 느꼈던 꾸준함의 가치를 잘 알고 있는 사람입니다. 이렇게 회사가 추구하는 인재상을 겸비한 저야말로 회사가 필요한 신입이라고 생각합니다.

1. **가장 자신있는 언어는?**

자바입니다. 처음 배운언어도 자바였고, 제가 해온 프로젝트들이 주로 자바에서 작업을 했던 프로젝트들입니다. Spring을 기반으로 게시판을 만드는 작업, 그리고 자바를 통한 안드로이드 어플 개발을 주로 했습니다.

**(기술 관련 질문)**

1. **무엇을 배웠는가?**

언어로는 자바, 파이썬, 자바스크립트등을 배웠는데 파이썬과 자바스크립트 같은 경우는 문법정도를 숙지한 정도이고, 주로 자바를 많이 했습니다.

자바를 통해 Spring기반 게시판만들기를 하고, 자바언어로 안드로이드 작업도 했습니다.

1. **프로젝트 할 때 어떤 내용을 다뤘는가?**

제 기준에서 크고작은 프로젝트를 진행했을 때, 2가지정도 말씀드리면

첫번째로 안드로이드를 활용한 자기계발 서비스 였습니다. 코로나로 인해 저하된 활동성을 다시끔 활성시키기 위해 사용자들에게 어떤 활동적인 퀘스트를 추천해주고, 사용자의 평점에 의해 추천알고리즘을 구현하여 개인화된 서비스를 진행하였습니다. 이때 제가 맡은 부분은 Firebase의 Ml kit를 사용하여 객체 인식 및 GPS를 통한 퀘스트 검증부분을 맡았습니다.

다음 말씀드릴 프로젝트는 크롤링을 통한 웹페이지 기능 구현 및 관리였는데요, 현재 운영하고 있는 사이트중 한 곳을 크롤링하여 그 페이지의 구조를 파악하고, 기능구현을 하며, DB를 설계하고 관리자페이지를 따로 제작하여 관리하는 프로젝트를 진행하였습니다. 이때 제가 맡은 부분은 관리자페이지를 설계하고 구현하는 부분이였습니다.

1. **프로젝트 할 때 API를 썼는가? 너가 담당 했는가?**

Firebase의 ML Kit를 사용했고 제가 담당했습니다. 약 1개월반의 시간동안 진행된 프로젝트 상 현실적으로 머신러닝을 새로 배우고 학습시키기에는 무리가 있다고 판단하여 이미 학습된 Firebase의 ML Kit를 적용하여 시간과 비용을 모두 잡았습니다. 사용한 머신러닝은 객체인식 Image Labelling이였습니다.

1. **Git에서 add, commit, push를 설명해주세요.**

간단하게 말하면 add를 통해(Staging area)에 commit을 진행하기 전 임시 저장을 하고, commit을 통해 push를 하기 전단계, 즉 git에 올릴준비를 합니다.

그리고 push를 통해 git에 파일을 올리고 추가적으로 pull을 통해 파일을 업로드 하게 됩니다.

**(자바 관련 질문)**

1. **오버로딩과 오버라이딩에 대해 말해주세요ㅔ**

먼저 **오버로딩**은 리턴타입, 함수명은 같은데 파라미터만 다른것이다.

주로 메소드 이름을 일관성 있게 하려고 하는 의도가 필요할 때 사용합니다.

**오버라이딩**은 재정의라고 표현할 수 있는데, 부모클래스의 메소드를 물려받아 자식크래스에서 재정의 하여 사용하는 것입니다.

1. **Stack queue에 대해 설명해주세요**

먼저 **stack**같은 경우는 후입선출이라고 표현할 수 있겠습니다.

데이터를 처리할 때 제일 나중에 입력된 데이터가 제일 먼저 출력이 되는 구조입니다.

예시로 ‘뒤로가기’ 버튼을 들면 되겠습니다. 여러 페이지를 방문하다가 뒤로가기를 누르게 되면 제일 나중에 방문한 페이지가 나오기 때문입니다.(last in first out)

다음 **queue**같은 경우는 선입선출이라고 표현할 수 있겠습니다. 마찬가지로 데이터를 처리할 때 제일 먼저 입력된 데이터가 제일 먼저 출력이 되는 구조입니다.

예시로 ‘터널’을 들면 되겠습니다.제일 먼저 터널로 들어간 차가 제일 먼저 밖으로 나오기 때문이죠(first in first out)

1. **접근제어자에 대해 설명해보세요**

객체지향 언어의 특징중 하나인 은닉화를 보여주는 개념인데요,

사용자가 굳이 알 필요가 없는 정보는 사용자에게 보여져서는 안되기 때문에 사용합니다.

자바로 예를 들면 private, public, default, protected이렇게 4가지로 구성.

Private는 같은 패키지 내에서도 같은 클래스에서 접근 할 수 있고,

Defalut는 같은 패키지 내에서 접근 할 수 있습니다.

Protected 같은 패키지와 추가적으로 외부 패키지 중 상속받은 서브클래스에서 접근 할 수 있고

마지막으로 public은 프로그램의 어디서나 접근 할 수 있습니다.

이때 접근제어자를 명시하지 않으면 기본적으로 public의 형태를 띄게됩니다.

1. **list 와 map과 같은 객체특징을 서술하시오**

둘의 공통점은 데이터를 저장하는 자료구조이다.

list같은경우 내부구현이 배열로 되어 있고, 저장되는 데이터의 포지셔닝이 의미가 있을 때 사용하고 데이터의 저장속도가 빠르다.

Map같은 경우 빈공간을 찾아 데이터가 저장되는 시스템이기에 상대적으로 데이터의 저장속도가 느리다. 특징으로는 key와 value값을 쌍으로 이루어 이용한다.

1. **Abstract, Interface(추상클래스, 인터페이스)에 대해 설명하시오**

Abstract를 말씀드리기 앞서 추상메소드와 추상클래스를 설명드리려고 하는데요, 추상메소드는 비유하자면 빈 껍데기라고 생각합니다. 메소드는 정의 되어 있지만 내용이 없는 비어있는 메소드 입니다.

추상클래스를 사용하는 이유는 여러가지가 있겠지만 적어도 추상클래스를 상속받은 클래스는 추상클래스에 있는 모든 추상 메소드를 구현해야 합니다.

이때 인터페이스와 추상클래스의 차이를 말씀드리면 **첫번째로** 인터페이스는 모든 메소드가 추상메소드인 반면 추상클래스는 추상메소드와 일반 메소드가 공존 할 수 있습니다.

추가적으로 인터페이스는 틀을 제공한다고 보면 되겠습니다. 즉 설계도 처럼 이렇게 만들면 된다 라고 규칙을 정해주는 것이라고 보면 되겠습니다.

반면 추상클래스는 부모클래스로서 의 용도로 쓰이기 때문에 기능의 확장성을 위해 사용합니다. 추가적으로 인터페이스는 다중상속이 가능하다.

1. **Spring Framework의 AOP, POJO, DI에 대해 설명해보세요**

**AOP**란 Aspect-Oriented Programming의 약자로 관점 지향 프로그래밍이라 불림. 비즈니스 로직 ,인프라 로직 즉 핵심적인 부분과 부가적인 부분을 나누고 그 관점을 기준으로 모듈화 하는것입니다. 이가 필요한 이유는 인프라 로직은 특수하게 어떠한 부분에서 핵심적인 것이 아닌 전 영역에 필요 할 수 있고, 이에 따라 중복될 수 있으며 이에 따라 유지보수가 힘들어 지기 때문입니다. 그렇기에 이 둘을 분리하여 재사용하는 것이 AOP의 취지이다. 주로 Been에서 사용됨.

**POJO**란 ?

**DI란** dependency injection의 약자로 의존성 주입이라고 불립니다.

쉽게 비유해서 설명하기위해 핸드폰 배터리의 일체형과 조립형을 예시로 들고 싶은데요, 이때 DI는 조립형이라고 보면 되겠습니다. 집적도 면에서는 떨어질 수 있어도 다른것으로 갈고, 업데이트 하기 용이하기 때문에 이 DI를 사용하는데 스프링에서는 크게 setter주입과 생성자 주입이 있습니다.

(예시 좀 알아볼까)

1. **객체지향 언어의 특징 5가지는?**

캡슐화 객체의 변수 및 메소드를 외부 객체가 함부로 접근하지 못하도록 하는 것

은닉화 (= 캡슐화)

추상화 공통의 속성,기능을 묶어 이름을 붙이는 것.

추상클래스, 인터페이스를 통해 구현된다.

상속성 상위클래스의 모든 속성과 연산을 하위클래스가 물려 받는 것.

다형성 하나의 객체가 여러가지 타입을 가질 수 있다

한 레퍼런스 변수가 다른 형태의 객체를 참조할 수 있다.

오버로딩, 오버라이딩, 업캐스팅, 다운캐스팅 등의 방법이 있다.

1. **Maven에 대해 설명할 수 있는가?**

라이프 사이클 관리하는 도구.

1. **Servlet.xml이 어떤 로직으로 돌아가는지 알려 줄 수 있는가?**
2. **get post의 차이점은?**

클라이언트가 서버로 요청을 보내는 방법의 2가지.

Get – 어떠한 정보를 가져와 조회하기 위해 사용되는 방식

url에 변수를 포함시켜 요청. 그만큼 보안에 취약

post – 데이터를 서버로 제출하여 추가 또는 수정하기 위해 사용

url에 변수를 노출안함. 상대적으로 보안에 유리

1. **비동기식과 동기식 차이**
2. **가비지 컬렉션이란**
3. **Vector와 ArrayList의 차이점은?**
4. **자바의 메모리 영역은?**
5. **제네릭이란?**

클래스에서 사용할 타입을 클래스 외부에서 설정하는 것.

1. **API란?**

Application Programming Interface

API는 프로그램들이 서로 상호작용하는 것을 도와주는 매개체.

이를 사용하면 구현 방식을 알지 못해도 제품, 또는 서비스가 서로 커뮤니 케이션 할 수 있고, 애플리케이션 개발을 간소화하여 시간,비용을 절약할 수 있다.

1. **라이브러리란?**

**(프로그램을 개발하기 위해 쓰는 공구와 같은 도구들)**

재사용이 가능한 필요기능으로 반복적인 코드 작성을 없애기 위해 언제든지 필요한 곳에서 호출하여 사용할 수 있도록 Class나 Function으로 만들어진 것.

1. **프레임워크란? Ex) 스프링**

**특정 프로그램을 개발하기 위한 여러 요소들과 매뉴얼인 룰을 제공하는 프로그램**

어떤 프로그램을 만들기 위한 기본 틀.

소프트웨어의 구체적인 부분에 해당하는 설계와 구현을 재사용이 가능하게 끔 일련의 협업화된 형태로 클래스를 제공하는 것.

장점 – 프로그램을 편리하고 빠르게 만들 수 있다.(생산성 향상)

체계적으로 만들어진다.(유지보수가 쉽다.)

1. **자바에 대해 설명해바**

절차지향 언어(순차적으로 실행되는 언어)가 아닌 객체지향 언어. 운영체제에 종속되지 않고 독립적으로 사용 가능한 언어.

객체지향 언어의 특성상 코드 작성의 재사용의 원활, 유지보수에 용이

가비지 컬렉터에 의한 효율적인 메모리 관리

**(이력서, 포트폴리오 관련)**

* + - 1. **고졸이네요? 왜 대학교는 안가셨어요?**
* 중학교부터 배드민턴 선수를 준비했다. 그때 미련하게 공부는 저버린채 운동에만 열심히 하다가 고등학교 2학년때와서 대학교의 현실적인 입시가 느껴졌고, 그때부터 수능을 준비했다. 체대쪽으로. 3수까지 했는데 원하는 대학교에 비해 점수가 잘 안나와 상심이 컸다. 바로 군대로 빤스런.
* 작년 12월에 전역을 하고 바로 IT쪽으로 발을 내딛였고, 현재 학점은행제를 통한 학사취득 준비중이다.
  + - 1. **Sl아카데미 AI를 활용한 빅데이터 분석. 뭘 배우셨나요?**
* 사실 AI와 빅데이터에 관한 수업은 거의 없었던 것 같습니다. 주로 JAVA의 기초부터 시작해서 Spring, 그리고 웹 DB를 통해 게시판 만들기를 배웠습니다.
* 그리고 2주마다 한번씩 팀을 짜서 팀별 프로젝트를 진행했는데, 그때 어떤 언어를 새로 팀끼리 학습하거나, 배운걸 토대로 응용하여 무언갈 만들어 보았는데, 이때 새롭게 무언가를 스스로 배워 나가는 어떠한 태도, 그리고 팀과 함께 협엽하고 소통하는 능력을 많이 배운 것 같습니다.
  + - 1. **상속구조를 활용한 프로그래밍이 무엇?**
      2. **Generic을 활용한 프로그래밍이 무엇?**
      3. **프로젝트 하면서 제일 힘들었던점?**
* 아무래도 팀원의 연락부재 및 미참여 였던 것 같다.

팀원 각각에게 부여된 일이 있고, 매일 점검하면서 프로젝트를 진행하고 있었는데, 어느순간 팀원 중 한명이 매일 어떠한 핑계를 대며 참여를 안하다가 결국 그뒤로 끝까지 참여하지 않았다. 그당시 팀장이였는데, 이를 빠르게 다른 팀원들과 상의해서 임무분담을 다시하고, 해결했다. 근데 그 뒤로 참여안하던 그 팀원이 다른 팀이 되어 끝까지 프로젝트를 참여하는 모습을 보고 팀장으로서 뭔가 내가 끝까지 챙기지 않아서 그랬나 라는 생각이 들며 반성하는 계기가 됐다.

* + - 1. **Level UP프로젝트 할 때 머함?**
      2. **크롤링사이트 할 때 머함?**
      3. **배드민턴은 왜 그만뒀나?**
      4. **수색대는 왜갔나?**
      5. **왜 개발자를 선택했나?**

**(보유기술)**

1. **Java의 수준은 어느정도?**
2. **HTML, CSS, JavaScript의 수준은 어느정도?**
3. **Python의 수준은 어느정도?**
4. **Android의 수준은 어느정도?**

**DBMS : DataBase Management System – db를 관리하는 도구**