

[Section 3] 기술면접 가이드

시간 약속은 철저히 지켜주셔야 합니다.

- 부득이한 상황이 발생한 경우, 미리 페어와 대화하셔서 시간 조정을 해주세요.

아래 기술문제 번호와 영상을 올리는 표 이외의 부분은 절대 임의로 편집하거나 수정하지 말아주세요.

- 혹시 수정이 필요한 부분이 발견된다면, 이슈슈어링을 통해 공유해주시면 빠르게 조치하겠습니다.

1. chapter / 기술면접 준비 시간에 각자 답하고 싶은 기술 면접 질문을 선택하고, 해당 질문들을 위한 답변을 준비합니다.

주어진 시간(2인 페어 기준 30분)을 고려하여 아래 리스트에서 내가 답하고 싶은 기술 면접 질문을 선택합니다.

선택한 기술면접 질문의 번호를 아래 표의 자신의 이름에 해당하는 열의 **기술 문제 번호**에 기입합니다.

예를들면, 만약 내가 **페어1** 이고 1,2,3번의 면접 질문을 선택한 경우, 아래 표에서 나의 이름이 있는 위치에서 **기술 문제 번호(페어1)**를 찾아 1,2,3번을 기입합니다.

기술 면접 질문을 선택하고 표에 기입했다면, 남은 시간 열심히 기술면접을 위한 준비를 합니다.

아래 표의 간단 예시를 참고해주세요.

API 계층

1번 - Spring MVC 프레임워크의 요청처리 과정에 대해서 설명해 주세요.

2번 - Spring MVC에서 제공하는 CSR(Client Side Rendering)과 SSR(Server Side Rendering) 방식에 대해서 설명해 주세요.

3번 - Spring MVC에서 REST API 엔드포인트를 구현하기 위해 사용되는 애너테이션들에 대해서 설명해 주세요.

@Controller, @RestController, @RequestMapping, @GetMapping, @PostMapping, @PatchMapping, @DeleteMapping 등

4번 - Controller에서 응답 객체로 사용하는 ResponseEntity에 대해서 설명해 주세요.

ResponseEntity 의 장점

5번 - Rest Client가 무엇인지 설명해 주세요.

6번 - Spring 에서 사용하는 Rest Client에 대해서 설명해 주세요.

7번 - DTO가 무엇인지 설명해 주세요.

8번 - DTO의 장단점에 대해서 설명해 주세요.

9번 - DTO 클래스의 Request Body에 대한 유효성 검증 방법에 대해서 설명해 주세요.

예외 처리

10번 - Spring MVC에서의 예외 처리 기법에 대해서 설명해주세요.

Controller내에서의 @ExceptionHandler 상용

글로벌 Exception 처리

비즈니스 로직 Exception 처리

데이터 액세스 계층

11번 - ORM이 무엇인지 설명해 주세요.

12번 - Spring JDBC, Spring Data JDBC, Spring Data JPA의 차이점을 설명해 주세요.

13번 - Spring Data JDBC에서의 애그리거트와 애그리거트 루트에 대해서 설명해 주세요.

14번 - JPA에서 의미하는 Persist에 대한 의미를 설명해 주세요.

15번 - JPA의 영속성 컨텍스트에 대해서 설명해 주세요.

16번 - JPA의 단일 엔티티 매핑 방법에 대해서 설명해 주세요.

@Entity, @Table, @Id, @GeneratedValue, @Column 등

17번 - JPA의 엔티티가 연관 관계 매핑에 대해서 설명해 주세요.

17번 - JPA의 엔티티 간 연관 관계 매핑에 대해서 설명해 주세요.

@OneToOne, @ManyToOne, @OneToMany 등
mappedBy 속성에 대한 설명

트랜잭션

18번 - 트랜잭션이 무엇인지 설명해 주세요.

19번 - 트랜잭션 경계가 무엇인지 설명해 주세요.

20번 - Spring에서 트랜잭션을 설정하는 방법에 대해서 설명해 주세요.

@Transactional 애너테이션 방식

AOP 방식

21번 - 분산 트랜잭션이 무엇인지 설명해주세요.

22번 - 트랜잭션 전파(Transaction Propagation)가 무엇인지 설명해 주세요.

23번 - 트랜잭션 격리 레벨(Transaction Isolation Level)이 무엇인지 설명해 주세요.

테스팅

24번 - 기능 테스트, 통합 테스트, 슬라이스 테스트, 단위 테스트에 대해서 설명해 주세요.

25번 - JUnit의 Assertion이 무엇을 의미하는지 설명해 주세요.

26번 - Given - When - Then 패턴에 대해서 설명해 주세요.

27번 - MockMvc를 이용해 Spring MVC의 API 엔드포인트인 Controller를 테스트하는 방법을 설명해 주세요.

28번 - Mock이 무엇인지 설명해 주세요.

29번 - Stub과 Stubbing이 무엇인지 설명해 주세요.

30번 - Mockito가 무엇인지, Mockito의 역할에 대해서 설명해 주세요.

31번 - @SpringBootTest와 @WebMvcTest의 차이점을 설명해 주세요.

API 문서화

32번 - Swagger와 Spring Rest Docs의 장단점을 설명해 주세요.

33번 - Spring Rest Docs의 API 문서화 동작 방식에 대해서 설명해 주세요

요.

34번 - Spring Boot 기반 애플리케이션을 빌드하는 방법에 대해서 설명해 주세요.

35번 - Spring Boot 기반 애플리케이션 빌드 시, 주로 사용하는 프로파일 (Profile)에 대해서 설명해 주세요.

36번 - Spring Boot 애플리케이션 실행 파일을 배포하는 방법에 대해서 설명해 주세요.

전통적인 배포 방식

클라우드 서비스에 배포하는 방식

API 계층

1번 - Spring MVC 프레임워크의 요청처리 과정에 대해서 설명해 주세요.

2번 - Spring MVC에서 제공하는 CSR(Client Side Rendering)과 SSR(Server Side Rendering) 방식에 대해서 설명해 주세요.

3번 - Spring MVC에서 REST API 엔드포인트를 구현하기 위해 사용되는 애너테이션들에 대해서 설명해 주세요.

@Controller, @RestController, @RequestMapping, @GetMapping, @PostMapping, @PatchMapping, @DeleteMapping 등

4번 - Controller에서 응답 객체로 사용하는 ResponseEntity에 대해서 설명해 주세요.

ResponseEntity 의 장점

5번 - Rest Client가 무엇인지 설명해 주세요.

6번 - Spring 에서 사용하는 Rest Client에 대해서 설명해 주세요.

7번 - DTO가 무엇인지 설명해 주세요.

8번 - DTO의 장단점에 대해서 설명해 주세요.

9번 - DTO 클래스의 Request Body에 대한 유효성 검증 방법에 대해서 설명해 주세요.

예외 처리

10번 - Spring MVC에서의 예외 처리 기법에 대해서 설명해주세요.

Controller내에서의 @ExceptionHandler 상용

글로벌 Exception 처리

로그인 Exception 처리

비즈니스 로직 Exception 처리

데이터 액세스 계층

11번 - ORM이 무엇인지 설명해 주세요.

12번 - Spring JDBC, Spring Data JDBC, Spring Data JPA의 차이점을 설명해 주세요.

13번 - Spring Data JDBC에서의 애그리거트와 애그리거트 루트에 대해서 설명해 주세요.

14번 - JPA에서 의미하는 Persist에 대한 의미를 설명해 주세요.

15번 - JPA의 영속성 컨텍스트에 대해서 설명해 주세요.

16번 - JPA의 단일 엔티티 매핑 방법에 대해서 설명해 주세요.

@Entity, @Table, @Id, @GeneratedValue, @Column 등

17번 - JPA의 엔티티 간 연관 관계 매핑에 대해서 설명해 주세요.

@OneToOne, @ManyToOne, @OneToMany 등

mappedBy 속성에 대한 설명

트랜잭션

18번 - 트랜잭션이 무엇인지 설명해 주세요.

19번 - 트랜잭션 경계가 무엇인지 설명해 주세요.

20번 - Spring에서 트랜잭션을 설정하는 방법에 대해서 설명해 주세요.

@Transactional 애너테이션 방식

AOP 방식

21번 - 분산 트랜잭션이 무엇인지 설명해주세요.

22번 - 트랜잭션 전파(Transaction Propagation)가 무엇인지 설명해 주세요.

23번 - 트랜잭션 격리 레벨(Transaction Isolation Level)이 무엇인지 설명해 주세요.

테스팅

24번 - 기능 테스트, 통합 테스트, 슬라이스 테스트, 단위 테스트에 대해서

설명해 주세요.

25번 - JUnit의 Assertion이 무엇을 의미하는지 설명해 주세요.

26번 - Given - When - Then 패턴에 대해서 설명해 주세요.

27번 - MockMvc를 이용해 Spring MVC의 API 엔드포인트인 Controller를 테스트하는 방법을 설명해 주세요.

28번 - Mock이 무엇인지 설명해 주세요.

29번 - Stub과 Stubbing이 무엇인지 설명해 주세요.

30번 - Mockito가 무엇인지, Mockito의 역할에 대해서 설명해 주세요.

31번 - @SpringBootTest와 @WebMvcTest의 차이점을 설명해 주세요.

API 문서화

32번 - Swagger와 Spring Rest Docs의 장단점을 설명해 주세요.

33번 - Spring Rest Docs의 API 문서화 동작 방식에 대해서 설명해 주세요.

34번 - Spring Boot 기반 애플리케이션을 빌드하는 방법에 대해서 설명해 주세요.

35번 - Spring Boot 기반 애플리케이션 빌드 시, 주로 사용하는 프로파일 (Profile)에 대해서 설명해 주세요.

36번 - Spring Boot 애플리케이션 실행 파일을 배포하는 방법에 대해서 설명해 주세요.

전통적인 배포 방식

클라우드 서비스에 배포하는 방식

2. pair / 기술면접 실습 시간에 모의 면접을 각각 30분씩 번갈아 진행합니다.

(3명인 페어의 경우 20분씩 진행)

준비한 답변을 기반으로 페어와 함께 모의 면접을 실시합니다.

면접관의 역할을 담당하는 페어가 모의 면접 시작과 동시에 녹화()를 시작합니다.

특별한 사정이 없다면 줌으로 녹화를 진행합니다. 녹화와 관련한 사항은 여기 공식문서에서 확인해주세요.

최고 주 저채 변경으로 기초 2이 문제해 히이 2이 10분 제한으로 변경된

첫 번째 모의면접은 첫 번째 페어에 대한 첫 번째 모의면접이었습니다. 이에 따라 번거로우시겠지만 한 페어의 모의 면접이 끝나면 한번 끊고, 다시 zoom을 시작하여 다른 페어의 면접을 진행해주세요.

면접관을 담당하신 페어는 최대한 답변이 일회성으로 끝나지 않도록 꼬리질문을 통해 추가적인 답변을 유도해주세요.

모의면접이 모두 종료되는 각자 이 [링크](#)를 통해 페어에 대한 모의 면접 피드백을 작성합니다.

녹화는 페어의 면접을 상대방이 녹화하기 때문에 업로드시에는 상대 페어의 **영상** 에 공유 링크 또는 파일을 업로드해주세요.

예를 들면, 내가 페어1인 경우 녹화한 페어2의 영상을 **영상(페어2)** 에 공유해주세요.

3. 녹화한 파일을 아래 노션 테이블에 업로드합니다.

4. 녹화된 영상을 다시 확인하면서 좀 더 보완할 부분들을 고민해봅니다.

녹화된 영상과 더불어 페어의 피드백을 함께 확인해보세요.

녹화 영상과 피드백을 확인하면서 느꼈던 부분을 블로그 등에 간단하게 기록해보는 것도 회고의 좋은 방법이 될 수 있습니다.

좀 더 보완해야 할 부분을 어떻게 채워갈 지 나만의 학습 계획 및 전략을 다시 고민해보세요.

