

데브파이브 대면 지필 고사

이름:

전화번호:

각 문제에 대하여 성실히 답해주시면 되며, 오답을 작성할 시 점수가 차감될 수 있습니다.

지필 고사를 보는 지원자는 해당 문제를 2025년 5월 1일까지 유출할 수 없습니다.

각 문항은 최대한 자세히 서술하고 왜 해당 답변이 나오게 되었는지 근거 혹은 예시까지 작성해주시기 바랍니다.

1. nodejs 가 무엇인지 서술하시오. (정답률 37%)

javascript 를 실행시키는 런타임, 실행기라는 표현

typescript를 실행할 수 있다는 표현, 최신 표준에서 실행 기능으로 포함되었으므로 인정
(자바스크립트를 실행한다가 기본 포함되어야 함)

2. 2025년 4월 기준 최신 리액트 버전을 적고, 메이저 버전 기준, 이전 버전과 어떤 점이 차이가 나는지 적으시오. (ex) 10.0.0 -> 11.0.0 에 대한 차이 (O), 10.0.0 -> 10.1.0 에 대한 차이 (X)) (정답률 16%)

18 -> 19 major 버전만 바뀌어도 정답처리

추가된 hook에 대한 서술이 있으면 인정

react compiler 에 대한 언급 인정

3. babel이 무엇인지 서술하시오. (정답률 41%)

다른 언어를 javascript 로 바꾼다는 표현 혹은 javascript 문법 버전을 변환한다는 표현, 번역기 인정
pt 로 바꾼다는 표현 혹은 javascript 문법 버전을 변환한다는 표현, 번역기 인정, 트랜스파일러 인정

4. nextjs에서 babel 이 사용되는지 서술하시오, 사용이 된다면 어떤 부분에서 사용되는 것을 알 수 있었는지, 사용되지 않았다면 그 대체재 혹은 대체 방법에 대하여 서술하시오. (정답률 8%)

사용된다 -> 해당 주장이 맞는지 검증, next config에 따라 포함될 수 있음

사용되지 않는다 -> SWC 에 대한 언급 유무 확인

babel을 사용한다고 정의하면 babel을 사용, next.config에서 아무것도 정의하지 않으면 swc 사용
(next v12부터 swc가 default)

5. nextjs 에서 server 컴포넌트와 client 컴포넌트의 차이를 서술하시오. (정답률 33%)

서술 확인, 클라이언트 컴포넌트가 서버에서 렌더링이 안된다 서버 컴포넌트는 된다는 것은 오답처리

SEO가 서버 컴포넌트가 더 유리하다는 것도 물론 오답처리

6. 다음은 nextjs 15 app router 프로젝트 내에 정의된 컴포넌트 파일이다. 이 파일은 정상 작동하지 않는데, 정상작동이 불가능한 이유와 어떻게 작동이 가능하게 할 수 있는지 서술하시오. (정답률 54%)

```
import { useState } from 'react'

export function Demo() {
  const [a, b] = useState(true)
  return <div onClick={() => b(false)}>hello</div>
}
```

“use client” 의 누락

*실 제본시 } 가 맨 뒤에 빠져있었음, 해당 부분을 언급해도 정답처리(시험 응시자중 1명이 존재했음 -> use client 와 함께 작성했으므로 어차피 정답처리)

7. 만약 위 컴포넌트가 소스코드 변화가 전혀 없이 정상작동할 수 있는 특정한 상황이 있다면 어떤 상황인지 서술하시오. (정답률 33%)

클라이언트 컴포넌트에서 호출 혹은 렌더링 되면 하위 컴포넌트들은 모두 클라이언트 컴포넌트가 되므로 작동함

8. nextjs 에서 ssg와 ssr의 차이를 서술하시오. (정답률 33%)

서술 확인, FOUC 현상 언급 오답처리 nextjs 에서의 ssr 모드이므로 해당되지 않음

9. **Suspense** 컴포넌트에 대해 서술하고 **nextjs** 에서의 **suspense** 사용과 일반 **react(spa)** 환경에서의 차이가 무엇이 있는지를 중심으로 서술하시오. (정답률 12%)

서술 확인, **nextjs**와 **react**의 **suspense** 차이가 서술되지 않으면 오답처리

10. 재귀함수에 대하여 서술하고 피보나치 수로 예시를 드시오. (정답률 45%)

서술 확인, 피보나치 함수가 작성되어 있을 경우 실제로 작동하는지 여부 확인

11. **benchmark** 의 정의에 대하여 서술하고, 이를 어떻게 **react** 에서 측정할 수 있는지 서술하시오. (정답률 8%)

성능 측정 관련 서술 + **react profiler** 언급, **lighthouse** 만 언급은 오답처리, **lighthouse** 언급을 하게 되면 모든 성능 측정 툴이 정답처리 될 수도 있음

12. lighthouse 에서 점수를 측정하는 기준을 말하고 각 점수를 올리기 위하여 어떤 조치를 취할 수 있는지 서술하시오. (정답률 54%)

성능

접근성

권장사항

검색엔진 최적화

웬만하면 모든 최적화 방법 인정하는 방향으로...

현실적으로 모든 항목을 외울 수는 없음을 인정

13. 다음 대괄호 사이 빈칸을 완성하시오. (비어 있는 것이 맞다고 생각하면 공란으로 유지) (정답률 8%)

```
type A<T> = [ ] // reverse util type!  
const a:A<number> = "" // type: string  
const b:A<string> = 1 // type: number
```

1. T extends number ? string : number

2. Exclude<number | string, T>

그외 방법은 서술 검증

14. 빈칸을 채워서 n! 이 작동되도록 하시오. (비어 있는 것이 맞다고 생각하면 공란으로 유지) (정답률 8%)

```
const factorial = (n) =>  
  Array.from([ ])[  
    .reduce([ ], )];
```

서술 검증