

웹 개발자 양성과정 튜터링 보고서

1. 학습 활동 보고서

학습회차	2	학습일자 및 학습시간	2023년 3월 24일(금) (10:00~14:00)	총 4시간
튜티참석자 명단	강제구, 공소연, 김민정, 김정혜, 김태연, 민한결, 안정민, 윤영서, 이성훈, 이수빈, 전현정, 한별, 한수정 (총 13명, 전원 출석)			
출석 체크				
튜티 이름	출석	지각 (출석 시간 기재)	결석	유고 결석 (증빙 서류 첨부 필수)
강제구	o			
공소연	o			
김민정	o			
김정혜	o			
김태연	o			
민한결	o			
안정민	o			
윤영서	o			
이성훈	o			
이수빈	o			
전현정	o			
한별	o			
한수정	o			
전달 사항(교수님 전달 사항 등 전달 사항을 입력하세요)				
1. 출석 관련 제공지 <ul style="list-style-type: none"> - 출석은 일반적인 수업대로 진행이 되며, 이는 성적 산출에 반영이 됩니다. - 유고 결석 서류 등 증빙이 불가능한 경우에는 무단 결석 처리가 됩니다. 				

2. 과제 관련 공지

- 매주 과제 기한은 튜터링 전날 자정까지 입니다.
- 이번 주차 튜터링 과제는 선택사항이라고 말씀 드렸기 때문에, 진행하신 분들만 저에게 개인톡으로 진행하신 과제 내용 전달주시면 됩니다.
- 과제를 해오시면 튜터링 시간에 코드 리뷰를 진행할 수 있습니다. 이 점 참고하시길 바랍니다.

3. 보고서 관련 공지

- 과제, 보고서 관련해서는 매주 주말 중으로 정리해서 제공 드리겠습니다.
- 과제 관련해서 막히거나 궁금하신 부분 개인톡으로 편하게 질문주시기 바랍니다. 너무 공통되는 부분들에 대해서는 단톡방에 답변을 드리겠습니다.
- 매주 질문하신 내용을 카카오톡 단톡방 게시판에 댓글로 남겨주시면 보고서에 이름과 함께 질문한 내용과 답변 작성해드리겠습니다.

4. 튜터링 진행 공지

- 코드 리뷰 관련하여, 한 명당 30분 이상이 소요되기 때문에 매주 모든 튜터분들의 과제에 대해서 리뷰를 해드리긴 어렵습니다. 따라서, 튜터링 시간에 1~2명의 과제를 임의로 뽑아 리뷰를 하고 공통 분모 외의 부분에 대해서만 개인적으로 질문 주시면 답변드리겠습니다.
- 코드 리뷰에서 피드백이 나온 부분에 대해서는 두 번의 피드백이 나오지 않도록 노력해주시기 바랍니다. 코드 리뷰에서 언급드리는 부분은 실제로 제가 코딩할 때 전부 다 지키는 부분입니다. 여러분의 코딩 실력 향상에도 도움이 될 테니 피드백을 드린 부분에 대해서는 최대한 지키시고 실용성 등에 대해 이해가 안되는 부분이 있다면, 적용하지 않고 과제를 제출하시기 보다는 그에 대한 질문을 하시는 것을 추천드립니다.
- 튜터링에서는 의도적으로 프론트엔드 개발과 과제에 대한 모든 부분을 설명해드리지 않습니다. 그 이유는 튜터 분들의 서칭 능력과 학습 능력을 기르는 것을 도와드리기 위함입니다. 따라서, 과제나 공부를 하다가 막히시는 부분은 바로 질문하시기 보다는 충분한 서칭을 해보고 난 후 그럼에도 모르는 부분들을 질문하시는 것을 추천드립니다.

5. 기타 공지

- 제가 공지라고 언급한 부분에 대해서는 읽으시면 이모지 남겨주시면 감사드리겠습니다.

- 카카오톡을 업무용으로 사용하고 있기 때문에, 카카오톡 알림을 조정하지 않고 있습니다. 때문에, 정말 급한 연락이 아니라면 너무 늦거나 이른 시간대에 연락은 자제 부탁드립니다. 가급적 오전 10시 ~ 오후 10시에 질문 및 과제 제출을 해주시면 감사드리겠습니다.
- 이번 주차에 동일한 튜터링 분반에서 3~4명씩 조를 짜라는 강사님의 요청이 있기 때문에 3.25 자정까지 카카오톡 톡 게시판에 팀을 자율적으로 정하여 댓글을 남겨주시기 바랍니다. 만약 자율적으로 팀 선출이 되지 않는다면 사다리타기로 지정하겠습니다.

강의 복습 내용(금주 강의에서 다뤘던 내용 중 복습한 내용을 입력하세요)

1. 브라우저 렌더링

- 대체적으로 교양처럼 보면 될 것 같습니다. 세세한 부분을 정확히 알 필요는 없고 나중에 찾아봤을 때 이해만 될 정도로 정리하면 될 것 같습니다.
- 웹브라우저는 인터프리터처럼 작동하기 때문에, html의 중간에 javascript 소스 코드가 존재한다면, 초기 렌더링 속도에 큰 지장이 있을 수 있습니다.
- 웹 브라우저 개발자 도구에서 해당 브라우저의 성능을 직관적으로 알 수 있으며 이를 잘 사용하면, 문제 상황을 정확히 인식하고 해결하는데 도움이 될 수 있습니다. 만약 브라우저 렌더링 속도가 특정 부분에서 지나치게 느리다면 개발자 도구를 통해 리전 문제(서버의 위치와 실제 웹 브라우저를 실행하는 사용자의 PC의 위치 차이로 인한 지연), 에셋 크기 문제 등을 파악하고 대처할 수 있습니다.
- 추가적으로 요즘은 웹사이트 성능을 측정하는 도구들이 많이 개발되었습니다. 예를 들어, 대중적으로 사용하는 chrome extension인 lighthouse 등으로 크롤링 봇이 해당 웹 사이트를 어느정도까지 인식하는지와 브라우저의 최적화 점수 등을 분석하여 직관적으로 표현해줍니다. 이러한 툴들을 잘 사용하면 웹 브라우저 렌더링 속도와 SEO를 개선하는데 큰 도움이 됩니다.

2. Javascript 소개

- 프론트엔드 개발자를 지망한다면, Javascript의 동작 원리를 정확히 이해하는데 많은 시간을 투자해야 합니다. 프론트엔드의 소스 코드의 양은 매우 방대하기 때문에 Javascript의 동작 원리를 제대로 이해하지 못한다면, 특정 문제 상황에 대한 디버깅을 할 때마다 매번 많은 시간을 쏟아야 할 수 있습니다.
- 현재 jQuery를 사용하지 않는 가장 큰 이유는 의존성 관리가 명확하게 되지 않아, 협업에 큰 어려움을 겪기 때문입니다.

3. Javascript 실습

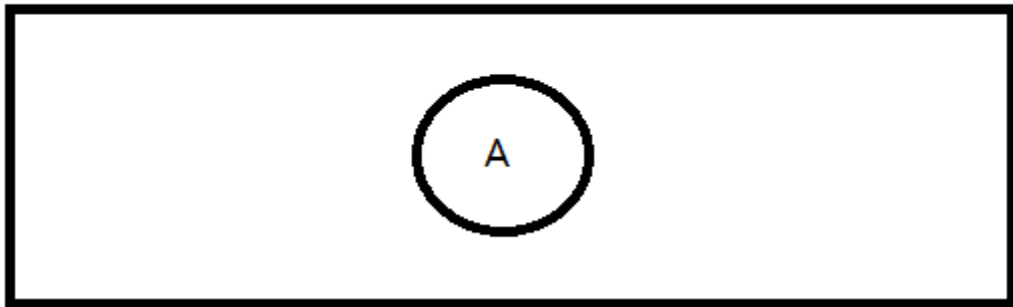
- window와 document는 웹 브라우저마다 기능의 미세한 차이가 존재합니다. 이는 cross browsing(요구된 다수의 웹 브라우저에서 정상적으로 실행되게끔 대응)에 대응할 때 고려해야 하는 사항입니다.
- javascript의 querySelector는 css의 selector와 비슷한 문법을 가지고 있습니다. 예를 들어 'div#wrap'은 태그가 div이면서 id가 wrap인 속성을 찾습니다.
- 버튼을 클릭했을 때 javascript 코드를 실행하는 html onclick property 를 사용했을 때, 주의 해야할 점은 html 파일은 위에서 아래로 한번만 읽힌다는 점입니다. 따라서 onclick에 함수를 할당했다면, 해당 함수가 onclick을 할당한 element의 위에 있는지, 아래에 있는지에 따라 다르게 동작할 겁니다.

튜티 질문 사항(금주 튜티들의 질문 사항과 답변 내용을 입력하세요)

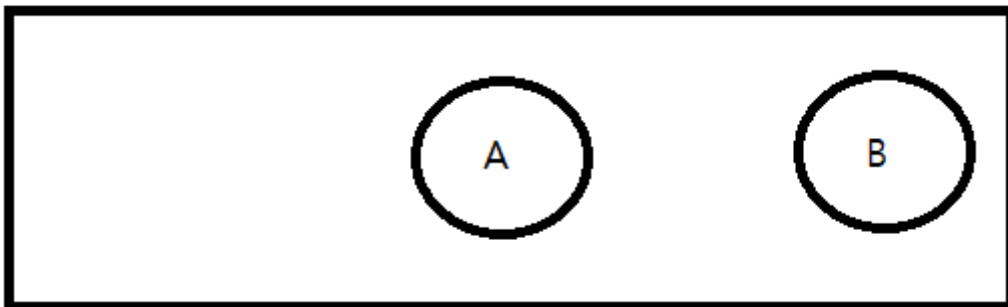
1. 이성훈 튜터

- FlexBox로 해결이 힘든 경우는 어떻게 해야 하나요?
 - FlexBox로 구현하지 못하는 레이아웃은 정말 일부분 특수한 경우를 제외하곤 거의 존재하지 않습니다.
 - 하지만, 무조건 FlexBox 만으로 구현을 하려고 하게 되면 소스 코드의 depth가 불필요하게 늘어나는 경우가 있습니다. 아래의 사진은 FlexBox로 구현했을 때 오히려 소스 코드가 복잡해질 수 있는 예시들 입니다.

1. margin: auto; 만으로 훨씬 간단하게 구현 가능한 경우



2. FlexBox로 구현하면 Depth가 불필요하게 많아지는 경우



- 따라서, 추후 유지보수 여부 등 상황에 맞게 FlexBox, margin, position 등을 사용하여 레이아웃을 구성하시길 바랍니다.
- 각각의 상황에서의 판단 기준은 프론트엔드 프로그래밍을 진행하면서 depth와 소스 코드를 최대한 줄이면서 작업을 진행하려고 노력할 때 생깁니다.
- 따라서, 빠른 작업을 하기 위해서는 이러한 케이스 들에 대응할 수 있게끔 많은 케이스에 대한 반복 작업을 통한 숙련과 코드를 개선하는 방식에 대한 끊임없는 고민이 필요합니다.

- ShoppingListData에서 ShoppingItem의 Type을 어떻게 지정하나요?

```

1  import React from "react";
2
3  type ShoppingItemType = {
4    title: string;
5    description: string;
6  };
7
8  type ShoppingListType = ShoppingItemType[];
9
10 const ShoppingItem: React.FC<ShoppingItemType> = ({ title, description }) => {
11   return (
12     <ul>
13       <li>제목 : {title}</li>
14       <li>설명 : {description}</li>
15     </ul>
16   );
17 };
18
19 const ShoppingListData: ShoppingListType = [
20   {
21     title: "가방",
22     description: "멋있음",
23   },
24   {
25     title: "신발",
26     description: "잘생김",
27   },
28 ];
29
30 const ShoppingList = () => {
31   return (
32     <ul>
33       {ShoppingListData.map((shoppingItemData, index) => (
34         <ShoppingItem
35           title={shoppingItemData.title}
36           description={shoppingItemData.description}
37           key={`shopping_item_${index}`}
38         />
39       ))}
40     </ul>
41   );
42 };
43
44 export { ShoppingList };
45
  
```

- 위의 예시를 들었을 때, 질문을 주신 부분입니다.
- 이 상황에서 ShoppingListType에 ShoppingItemType[]을 할당하였는데, 이는 {title: string; description: string;}[]을 할당한 것과 다른 의미입니다. 따라서, ItemType을 따로 선언하고 이에 대한 List를 ListType에 할당하는 것이 맞습니다.
- 런터에서 이러한 일부 상황들을 허용해주기도 하는 것 같습니다만, Type이 복잡해질수록 런터에서 인식하지 못하는 부분이 생기기 때문에 이러한 부분들을 충분히 이해하는 방식으로 코딩 공부를 하시는 것을 추천드립니다.

2. 민한결 튜터

- 리액트 컴포넌트 사용시, map 함수 사용을 어떻게 하면 되나요?
 - 위의 예시처럼 진행하시면 됩니다.
 - mdn 문서를 살펴보면, javascript의 map은 배열을 return하지만 작성하신 코드에서는 for문 만으로 구성되어 아무것도 return하지 않는 상황이라서 error가 나왔습니다.
 - 또한, 콘솔에서 length가 없다는 식의 내용이 나온다면 배열이 아닌 것에 반복문을 하진 않았는지 확인해보는 절차가 필요합니다.
- React를 localhost 실행하였는데, 하얀 화면만 나오면 어떤 문제인가요?
 - 하얀 화면만 나온다는 조건만으로는 문제 상황을 특정할 수 없습니다.
 - 따라서, 개발자 도구의 콘솔 부분을 참고하시어 버그를 트래킹하시기 바랍니다.

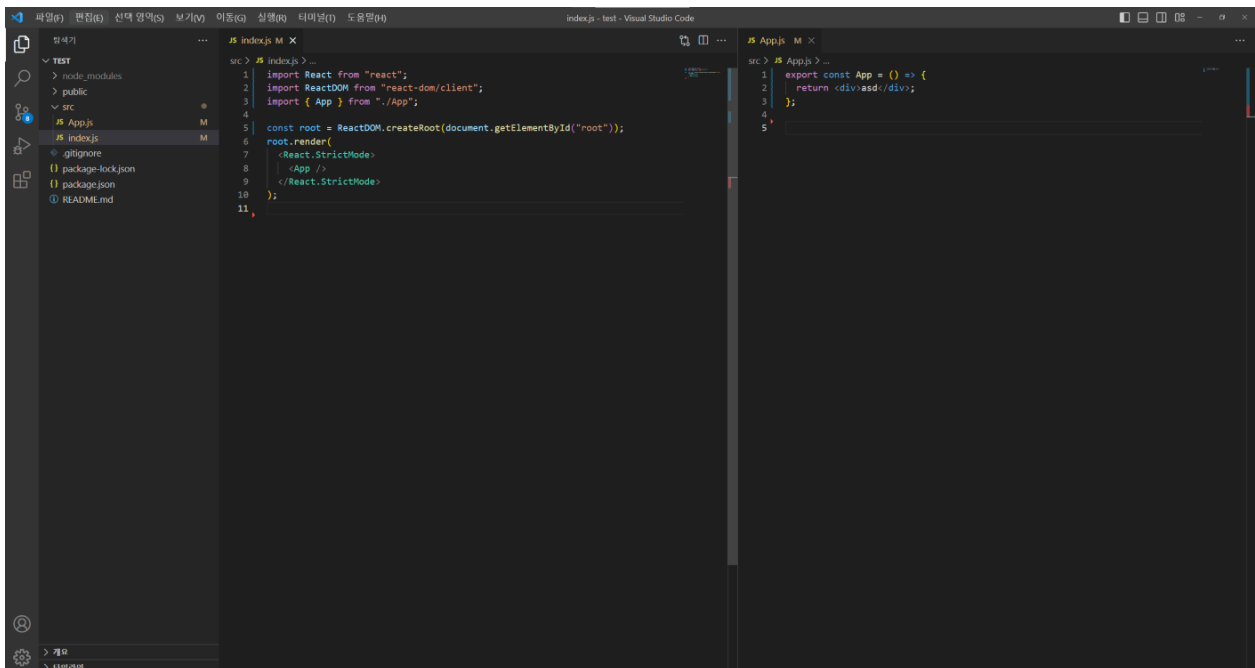
추가 학습 내용(강의 내용 외 튜터링을 하면서 다룬 내용들을 입력하세요)

1. React 개발 환경 설정

- 주요 명령어

기능	명령어	
CRA 레포지토리 생성	<code>npx create-react-app [project-name] -y</code>	
CRA 테스트모드 실행	<code>npm start</code>	
Module	<code>npm i [module-name]</code> 또는 <code>npm install [module-name]</code>	
vscode로 폴더 열기	<code>code [상대경로 or 절대경로]</code>	

- 초기 세팅 화면 (src 이하 파일만 조작)



2. github 초기 세팅 (vscode 연동, repository 생성 및 연동)

- 이 부분은 튜터링에서 초기 세팅을 하는 방법에 대해 한 번 다루었고, 막히는 튜티분이 없다고 하셨기 때문에 과정에 대해서 자세히 서술하진 않겠습니다.

과제물 내용(튜티들의 과제 결과물은 따로 첨부하세요)

[과제 공지, 선택이지만 진행하고 오시는 것을 권장 드립니다.]

3주차에서는 React에 대한 소개를 드리고 4주차에 React에 대한 기본적인 활용을 다룰 예정입니다.
따라서, 이번 주차의 과제는 다음과 같습니다.

1. flexbox froggy (<https://flexboxfroggy.com/>) 끝까지 해오기
2. bootstrap pricing page (<https://getbootstrap.com/docs/4.0/examples/pricing/>) 퍼블리싱 (HTML, css만 사용)
3. React 개발 문서를 통한 기본적인 React 개발 환경 설정 (<https://ko.reactjs.org/docs/getting-started.html>)
4. (2)에서 구현한 내용을 jsx 문법에 맞게 React 프로젝트에 옮기기 (선택)

튜터링의 진행은 정확히 정해진 바가 없습니다. 저는 아직 튜터분들의 정확한 웹 프론트엔드에 대한 지식 수준을 모릅니다. 따라서, 과제가 너무 부담스럽다거나 진행하기 어려우면 튜터 분들이 저에게 피드백을 주시면 여러분이 따라올 수 있게끔 반영하겠습니다.

다만, 튜터분들이 적극적으로 참여해주시면 참여해주시는 만큼 실질적으로 얻는게 더 많아진다고 생각해주시면 될 것 같고 헛된 시간은 쓰지 않으시도록 하겠습니다.

결론은 최대한 할 수 있는 만큼 해오시면 진행된 정도에 따라 앞으로의 과제 난이도를 조정하겠습니다.

--- 과제 피드백 ---

이수빈 튜터

[질문 사항]

1. header 태그에 할당된 css가 작동하지 않았던 이유

- style.css 보다 default.css를 이후에 불러왔고 default.css에서 header의 display를 block으로 할당했기 때문에, flexbox가 작동하지 않았던 것으로 보입니다.

[피드백]

1. css convention

- 전반적으로 naming을 고르게 사용하려고 노력하신게 보입니다. 하지만, 웹 개발에서 권장하는 묶음의 단위는 복수의 경우 container, 단수의 경우 wrapper입니다. 예를 들어, card-list는 card-container 라고 명명하시는게 조금 더 적절하겠습니다.

- css property 선언의 순서가 조금 뒤죽박죽인데 css ordering convention에 대해서 서칭해보는 것도 좋을 것 같습니다.

- font-size의 경우 rem을 주시고 width의 경우 px를 주신 것 같습니다. 보통 혼용하지 않고 하나만 쓰는 편이며, 여러 의견이 많지만 최신 기준으로는 px 기준으로 작업하시는 것을 추천드립니다. 이에 대해서는 추후 다룰 예정입니다.

- 보통 html element에 css를 할당할 때, 태그'만'으로 할당하진 않습니다. 의존성 이슈가 있기 때문입니다.

- 공부를 하실 때에는, 모든 태그의 style을 없애지 마시고 진행하시는 것을 권장드립니다. 당장은 편하겠지만 태그들에 대한 공부를 하지 않으면 나중에 태그의 기본 style을 사용한 디자인 라이브러리나 프로젝트를 접할 때 어려움을 겪을 수 있습니다. 또한, 생각보다 기본 style이 용이한 경우가 몇가지 있기 때문에 이

부분은 프로젝트를 설계할 때 충분한 고려를 한 이후에 보통 customizing 하는 편입니다.

2. input[type=button] 사용 관련

- 해당 방식은 요소의 children이 그저 text일 경우 동일한 결과를 낼 수 있습니다. 하지만 몇몇의 경우 children이 그저 텍스트가 아닌 아이콘이나 기타 요소들이 들어가는 경우가 있기 때문에 input type property에 button value를 써서 button 처럼 사용하는 것은 지양하는 편입니다.

3. image 태그 관련

- image는 html 표준 태그로 "img"를 사용합니다. 아마 크롬이나 몇몇 브라우저에서는 image로 작성해도 동작하는 것으로 알고 있는데, cross browsing 이슈가 존재하기 때문에 img로 작성하시는 것을 권장드립니다.

4. semantic tag 사용 관련

- semantic tag에는 보통 style을 할당하지 않는 편입니다. 또한, 보통 기존 semantic tag를 고려하지 않은 작업물에 디자인에 영향을 받지 않도록 덧씌우는 방식으로 작업을 합니다.

- semantic tag를 적절하게 사용하는 방법에 대해 공부하는 것은 semantic tag 외 사항들을 모두 완벽하게 숙지한 후에 진행하시는 것을 추천드립니다.

5. prettier 사용 관련

- 아마도 제 생각에 prettier를 사용하지 않으시는 것 같습니다. prettier를 사용하는 이유는 가독성 때문입니다. 사용하지는 것을 권장드립니다.

- 2주차 보고서에 언급된 바와 같이 frontend 개발을 할때는 indentation의 tab size를 2로 맞춰놓고 하는 편입니다. 지금 당장은 실습 코드가 적기 때문에 체감이 안되시겠지만 복잡한 화면을 구성하게 된다면 html dom의 depth가 너무 깊어 가독성에 심각한 문제를 줍니다. 이런 부분도 고려하면 좋을 것 같습니다.

안정민 튜터

1. css convention

- ducks 패턴(부모 요소의 네이밍을 자식 요소 앞에 붙이는 형태)을 꼭 사용해야할지 충분히 고려해보시면 좋을 것 같습니다. 2주차 보고서에 있는 내용과 같이 css 내장 ">" 연산자를 통해 다른 요소들과 class name이 겹치더라도 같은 css를 받지 않게끔 해보면 css naming이 조금 더 깔끔해질 것 같습니다.

- _(언더바)와 -(하이픈)을 혼용하시는 것 같습니다. javascript에서 -(하이픈)은 뉘셈 연산이기 때문에 보통 _(언더바)를 사용합니다. 이 부분은 html, css 만 작업하는 과제이기 때문에 자율이지만 하나만 사용해야 하는 것은 자명하기 때문에 둘 중 하나를 선택해서 작업하시면 좋을 것 같습니다.

- css property 선언의 순서가 조금 뒤죽박죽인데 css ordering convention에 대해서 서칭해보는 것도 좋을 것 같습니다.

2. reset.css

- 아마 이 부분은 기본 태그에 있는 css property를 없애는 소스 코드를 웹에서 끌어오신 것 같은데 button에 있는 기본적인 부분 css property까지 전부 없애버려서 button이 button 같지 않은 느낌이 들 수 있을 것 같습니다. 이런 부분도 고려하시면 더욱 웹 사이트 같은 느낌이 날 것 같습니다.

3. id 사용 관련

- 아시겠지만, html의 id 속성은 html 문서에서 여러번 선언 되었을 때, 가장 위에 위치한 element만 선택 됩니다. 이러한 부분을 인지하시고 작업하시면 좋을 것 같다는 생각이 듭니다. 때문에, 단순히 css만 지정하는 부분에서의 사용은 지양하는 편입니다.

4. prettier 사용 관련

- 아마도 제 생각에 prettier를 사용하지 않으시는 것 같습니다. 그렇게 생각한 이유는 index.html의 head에 있는 link가 닫히지 않은 상태로 있는데 인지하지 못하신 것 같아서 입니다. prettier를 사용하는 이유는 가독성 때문입니다. 사용하게 된다면 닫히지 않는 태그가 있을 때 바로 알 수 있기 때문에 사용을 권장드립니다.

5. indentation 관련

- 2주차 보고서에 언급된 바와 같이 frontend 개발을 할때는 indentation의 tab size를 2로 맞춰놓고 하는 편입니다. 지금 당장은 실습 코드가 적기 때문에 체감이 안되시겠지만 복잡한 화면을 구성하게 된다면 html dom의 depth가 너무 깊어 가독성에 심각한 문제를 줍니다. 이런 부분도 고려하면 좋을 것 같습니다.

참고 문서(참고한 문서나 문헌을 입력하세요)

<https://ko.reactjs.org/docs/getting-started.html>

<https://developer.mozilla.org/en-US/docs/Web/CSS/Specificity>

* 튜터링에서의 대부분의 설명은 mdn 문서를 기반합니다.

기타

이번 주차는 튜터링 팀 재배치로 인해 인원이 바뀐 첫 수업이었기 때문에, 앞선 내용들을 보고서를 바탕으로 리뷰했습니다. 전달드릴 내용과 실시간으로 판단해야 하는 부분이 많았기 때문에 호흡이 짧았고 쉬는 시간이 거의 없게 진행하였습니다.

활동 평가(자신의 평가를 입력하세요)

전반적으로 의도했던 바를 튜티분들에게 잘 전달드렸고 앞으로도 괜찮을 거라고 생각합니다. 제가 임의로 새로 배정된 3분의 튜티들에게 이번 튜터링에 대한 피드백을 요청드렸습니다. 다음은 튜티 분들의 익명 피드백입니다.

(익명 튜티1) : 이번 튜터링을 한 번 진행하면서 느낀 것은 제대로된 튜터링을 받고 있다는 느낌을 받았습니다. 과제 역시 난이도가 높지도 낮지도 않은 적정 수준을 유지하고 있다고 생각합니다. 약간의 개선점이 있다면 학생들에게 질문을 많이 던져주시고 대답을 요하셔도 좋을 것 같습니다. 물론 많이들 모르기 마련이지만 틀릴수록 더 많이 배워간다고 생각하기 때문입니다.

(익명 튜티2) : 이번 튜터링의 좋았던 점은, 수업하시는 것을 공유해서 보기 쉽다는 점입니다. 실습에 문제가 생기면 바로 피드백을 받을 수 있어서 좋았습니다.

(익명 튜티3) : 주차별 강의 자료를 보면서 얘기를 해주셔서 복습을 한 번더 할 수 있었고 강의 자료에 없는 추가적인 내용들을 더 알 수 있어서 좋았습니다. 매주 튜터링 과제를 통해 더 깊은 웹프로그래밍 공부를 할 수 있을 것 같습니다. 진행한 과제를 보면서 피드백 하는 시간이 좋았습니다. 다른 튜티분 코드도 보면서 참고할 수 있었고 내 코드와 비교해볼 수 있었습니다. 그리고 코드 피드백 해주시는 게 나중에 개발자로서

작업을 할 때 큰 도움이 될 것 같습니다. 마지막 리액트 실습을 할 때 진행이 빠르다는 생각이 들었습니다. 제가 리액트는 처음이라 앞으로 공부를 더 해봐야 할 것 같습니다.

피드백과 튜터링에서의 튜티 분들의 호응을 통해 제가 판단한 내용은 강의 내용 리뷰와 과제, 코드 리뷰 등 튜터링 자체엔 어느정도 만족하시는 분위기인 것 같고 추가적으로는 튜티분들의 참여도를 조금 더 이끌어냈으면 한다는 식으로 이해했습니다. 이번 주차에서는 앞서 서술한대로 3주차이지만 새로 배정된 팀에 대해 튜터링을 진행했기 때문에 호흡이 많이 짧았어서 그렇게 느끼셨을 수 있다고 생각합니다. 다음 주차부터는 피드백을 반영하여 튜티 분들이 최대한 참여를 할 수 있게끔 진행하려고 합니다.

튜터링 사진(4장 이상 필수) 업로드



강의실 사진1



강의실 사진2



강의실 사진3



강의실 사진 4

자료 업로드

튜티 과제 결과물 첨부

2. 교수 피드백

내용