<3주차 업무 피드백>

개발 할 땐 항상 다른 사람의 입장에서 생각하기

아예 숫자가 안눌리도록 설계하는게 좋을 것 같음(이용자 관점)

전역변수는 가장 위에 선언하는 게 좋다.(가독성을 위해서)

리스트끼리는 비교가 안되므로 리스트는 반복문을 이용해서 비교하는 것

Json은 이를 이용해서 비교

Html 옵션을 string으로 넣는 것은 매우 좋지 않음(html 적인 혜택을 아예 받을 수 없음 string 단일 형태로 강제 입력의 느낌)(유지보수에 매우 좋지 않음)

반복문을 쓸데 없이 쓴 것을 줄이는 것은 좋으나 필요에 의해 반복문을 사용하는 것은 괜찮다.

이벤트 함수는 뒤에 event를 붙인다. Function(event)같은 형식으로 붙여준다.

함수와 이벤트함수의 구분을 위해서 붙여주는 게 좋다.(html호출용인지 javascript용인지)

Bem 네이밍 기법 “성격 – 목적’’

Play – containor

Play\_box—button

헤드 바디 안으로 넣어주기

코드 최적화(for while 문 사용해서 줄여주기)

이미 입력되어진 값은 비활성화

함수 줄여주기(중복함수)

던지기 안되도록 예외 막기

\*\* 클린 코드

<정의>

* 프로그래밍 언어를 쓸 때는, 잘 돌아가는 것 뿐만 아니라 잘 읽히게 작성하는 것도 중요함

<조건>

* 1. 의미 있는 이름을 작성하기
* 2. 코드를 가독성 좋게 쓰기(개행, 주석, 간결한 로직, 파일 분할 등)
* 3. 통일성 (같은 코드 다르게 작성하면 안됨)
* 4. 유지보수성 (중복 코드 제거, 뭐 하나 수정하면 다같이 바뀌는 것 등)
* 5. 테스트를 쉽게하기 (조건이 맞추어져 있는지 확실하게 판단이 가능한가)\

<의미 있는 이름을 짓는 방법>

* 세 번째 원칙에 포함되는 내용
* 1. 형태  
  카멜, 스테이크(변수나 함수이름), 파스칼   
  camel : loginButton (프로그래밍에 관련된 것)  
  snake: login\_button (프로그래밍에 관련 없는 것)  
  paskal : LoginButton(객체 이름을 지을 때)
* 2. 단어 선택  
  이름은 길어도 상관이 없다. 충분한 정보를 포함하자. 의미없이 줄여쓰지 말자  
  관례라면 상관이 없으나 나 혼자 쓰는 약어는 사용 지양  
  통일된 단어 쓸 것  
  개발자들이 많이 쓰는 단어 쓸 것  
  사전을 찾아야 되는 단어 쓰지 말 것
* 3. 단어 조합  
  변수 : 명사의 조합(목적, 요소)  
  함수 : 동사와 명사의 조합(기능, 결과)

<4주차 과제>

//말이 되는 프로그램

// 기능 예외 xu

퍼즐 게임

게임 시작하면 3x3퍼즐이 섞여서 출력 그림하나를 쪼개기

반대편에 3x3 빈공간이 있도록

Drag & drop으로 옮기고 맞으면 끝

시작지접, 정답지점

1. 시작 지점에서 드랍까진 드랍이 가능
2. 정답이 잘못되었으면 다시 시작 지점에 가져다 두기
3. 시작 지점 안에서 dd 가능 상호간에 위치 변경 가능
4. 드롭 지점 ‘’
5. 이미 있는 자리에 dd 하면 두개 위치 바꾸기

풀이 시간, 난이도 설정, 원하는 이미지를 받아서 잘라서 사용