



# 웹프로그래밍 프로젝트



10612 박형근



# 목차

1. 프로젝트 진행 이유
2. 주요 기능 소개
3. 기능 시연
4. 마무리



# 프로젝트 진행 이유

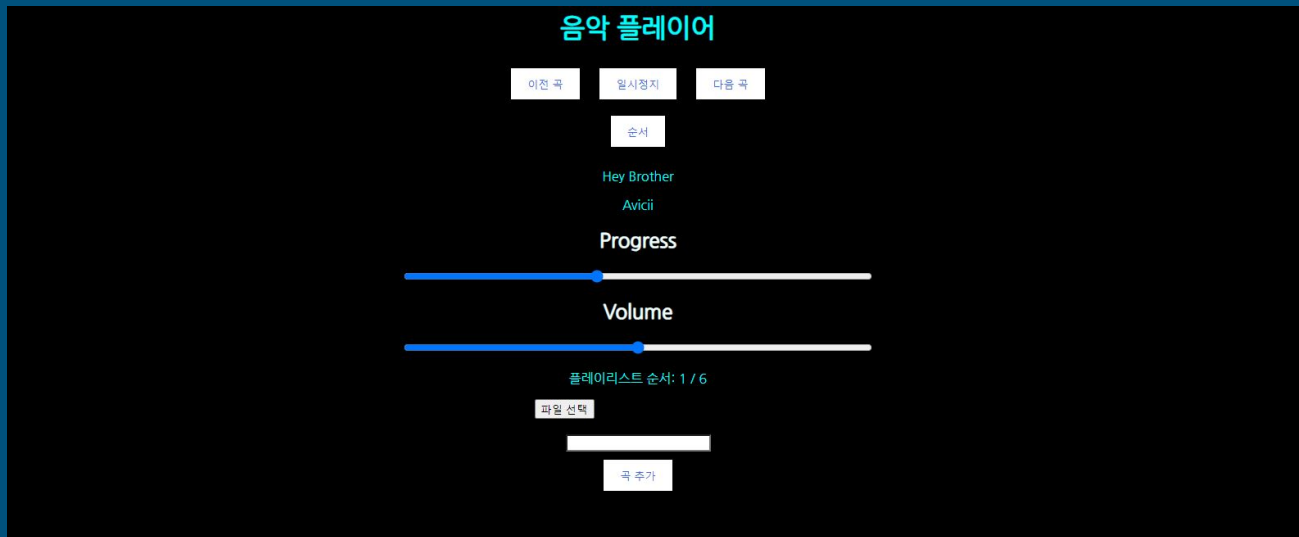
---

- 음악을 웹에서 재생할 수는 없을까?
- 왜 고정된 디자인을 사용해야 할까?



# 주요 기능 소개

## 주요 화면



# 주요 코드 소개

```
// 음악 '플레이리스트' 정보
const playlist : [{song: string, artist: string,...} = [
{
  song: 'Hey Brother',
  artist: 'Avicii',
  src: '../Avicii - Hey Brother.mp3'
},
{
  song: 'The Nights',
  artist: 'Avicii',
  src: '../Avicii - The Nights (Audio).mp3'
},
{
  song: 'Levels',
  artist: 'Avicii',
  src: '../Avicii - Levels.mp3'
},
{
  song: 'Waiting For Love',
  artist: 'Avicii',
  src: '../Avicii - Waiting For Love.mp3'
},
{
  song: 'Wake Me Up',
  artist: 'Avicii',
  src: '../Avicii - Wake Me Up (Official Video).mp3'
},
{
  song: 'The Days',
  artist: 'Avicii',
  src: '../Avicii - The Days (Lyric Video).mp3'
}
]
// ... (다른 곡 정보)
]; // 배열 안 사용자 객체로 곡 정보 저장
```

기본 음악 저장용 배열

```
// 파일 선택 시 플레이 리스트에 곡 추가하는 함수
1개의 사용 위치

function handleFileSelect(event) : void {
  const files : FileList | ... = event.target.files;
  for (let i : number = 0; i < files.length; i++) {
    const file = files[i];
    const track : {...} = {
      song: file.name.replace(/\.([^.]+)$/, ""), // 파일 이름으로부터 확장자 제거
      artist: document.getElementById( 'song-input').value, // artist는 직접 받아온다.
      src: URL.createObjectURL(file) // 파일의 URL을 트랙의 소스로 사용
    };
    playlist.push(track); // 플레이 리스트에 트랙 추가
  }

  // 추가된 곡들을 플레이 리스트에 반영한 후 첫 번째 곡 로드
  loadTrack(currentTrackIndex);
  updateOrderInfo();
  audioPlayer.play();
  document.getElementById( 'song-input').value = ''; //입력칸 초기화
}
```

음악 추가 시 사용되는 함수

# 마무리

---

## 소감

이 프로젝트를 진행하면서  
여러 가지를 배울 수 있었으며  
좋은 경험이 된 것 같습니다.

## 느낀 점

원래 계획이 매우 방대해서  
계획대로 하지 못하고 틀었던 것이 아쉽고,  
계획을 너무 크게 잡으면 안 된다는 것을  
느꼈습니다.