

# Persoonlijke spotify wrapped

Opdrachtgever: Ryan Putman

Opdracht: ik heb laatst bij spotify mijn account gegevens opgevraagd om dat ik geïnteriseerd was in de data die spotify bij hield en om te zien waar ik naar luister.

Bij het ontvangen van deze data stond het in json bestanden met weining tot geen overzicht op de data. Ik kon er in de huidige staat dus niks mee. Daarom kwam ik met het idee om een klein code programma te maken dat kon tellen hoe vaak elke artiest voor komt in de lijst. En de top 3 artiesten waar ik naar luister kan weergeven.

## Aanpak:

Omdat de data in Json bestanden zit moeten we deze er eerst uit halen om deze te gebruiken in javascript. Dit heb ik gedaan door middel van een fetch command.

```
function getdata(){
  fetch('./MyData/StreamingHistory' + x + '.json')
    .then(response => response.json())
    .then((data) => makearray(data));
}
```

Nu dat we de daten in Js hebben kunnen we gaan beginnen met kijken hoe vaak elke artiest voorkomt. Dit heb ik gedaan door middel van de “plays” functie deze functie kijkt door heel de data heen en kijkt of de data de waarde “artistName” bevat.

```
function plays(array, meting){
  var playarray = [];
  array.forEach((e)=>{

    if(playarray.some((val)=>{ return val[meting] == e[meting] })){
```

Voor alle data die een “artistName” heeft word gekeken of er al een nieuw object in de array is met occurrences.

```
    playarray.forEach((k)=>{
      if(k[meting] === e[meting]){
        k["occurrence"]++
      }
    })
  })
```

Zo nee dan word deze aangemaakt in de else statement. Als hij al wel bestaat in de nieuwe array dat word de occurrence met 1 omhoog gedaan.

```

}else{
  let a = {}
  a[meting] = e[meting]
  a["occurrence"] = 1
  playarray.push(a);
}

```

Nu hebben we hoe vaak een bepaalde artiest is afgespeeld maar het is nu nog in compleet random volgorde.

```

▶ 1: {artistName: 'b6ch', occurrence: 10}
▶ 2: {artistName: 'glaive', occurrence: 12}
▶ 3: {artistName: 'ericdoa', occurrence: 9}
▶ 4: {artistName: 'JELEEL!', occurrence: 10}
▶ 5: {artistName: '5 Seconds of Summer', occurrence: 2}
▶ 6: {artistName: 'Joji', occurrence: 6}

```

Om dit te ordenen heb ik de '.sort' command

```

arraydata = arraydata.sort(function(a, b) {
  return a.occurrence - b.occurrence;
});
arraydata = arraydata.reverse()

```

De function bij sort word gebruikt om te zorgen dat het word gesorteerd op waarde van het nummer. Hierdoor komt de array wel in omgekeerde volgorde van klein naar groot alleen dat probleem kan makkelijk opgelost worden met .reverse()

```

▶ 0: {artistName: 'CLUT', occurrence: 4007}
▶ 1: {artistName: 'Billie Eilish', occurrence: 995}
▶ 2: {artistName: 'Post Malone', occurrence: 603}
▶ 3: {artistName: 'Joost', occurrence: 445}
▶ 4: {artistName: 'Years & Years', occurrence: 400}
▶ 5: {artistName: 'Goldband', occurrence: 364}

```

Nu we het correct en in de goede volgorde in de console hebben kunnen we door middel van een for loop en "document.createElement()" het weergeven in html vorm.

```

for(var i = 1; i <=3; i++){
  const wrapped = frame.appendChild(document.createElement("div"))
  wrapped.classList.add("wrapped")
  const placement = wrapped.appendChild(document.createElement("p"))
  placement.innerText = [i] + "."
  const artist = wrapped.appendChild(document.createElement("p"))
  artist.classList.add("add")
  if(meting == "trackName"){
    artist.innerText = arraydata[i-1].trackName;
  }else if(meting == "artistName"){
    artist.innerText = arraydata[i-1].artistName;
  }

  const plays = wrapped.appendChild(document.createElement("p"))
  plays.classList.add("add")
  plays.innerText = "aantal plays: " + arraydata[i-1].occurrence;
}

```

eindresultaat:



Wat heb ik geleerd:

- Hoe ik arrays kan ordenen
- Werken met arrays
- Volgorde van arrays aanpassen