

JS: Partikkel-fest

Introduksjon

Denne oppgaven bygger på koden du skrev i oppgaven [Partikkel-anim](#) å gjøre [Partikkel-animasjon](#) før du fortsetter på denne oppgaven.

Her skal vi videreutvikle `partikkel`-animasjonen vår slik at den ser s



Merk at i denne oppgaven vil du kun få beskrevet hva du skal gjøre med koden.

Steg 1: Hva må gjøres?

I denne oppgaven får du kun små eksempler på kode for å hjelpe deg gjennom tankemåten til å lage animasjonen over ved å presentere en

La oss studere animasjonen og analysere hva den inneholder:

- ☐ Et partikkel i midten av skjermen som alltid er der. Hva kan være

- ☐ Partiklene som går ut fra midten og blir mindre og mindre jo len
- ☐ Hastigheten til hvert partikkel varierer
- ☐ Retningen varierer, men et partikkel reiser i en rett linje
- ☐ Det er mange partikler som blir til hvert sekund

La oss analysere punktene over, og se hva på hva vi må programmere

- ☐ Siden partiklene går ut fra midten må jo alle starte der, derfor n for hvert partikkel.
- ☐ Siden partiklene blir mindre og mindre, men starter med samme partiklet på samme måte som vi gjør når vi skal flytte på det. Ti effekt.
- ☐ Siden hastigheten varierer kan vi bruke `Math.random` til `xSpeed`

```
xSpeed: Math.floor(Math.random()*20 - Math.random()*20));
```

Dette vil gjøre at du får et positivt eller negativt tall med varierende samme for `y`-retningen for å få partiklene til å bevege seg overalt

- ☐ For å få dem til å følge en rett linje bruker vi bare endringer i `x`.
`particle.x + particle.xSpeed;`
- ☐ Siden det er mange som blir laget på engang må vi for hver gar og bruke en `for`-løkke til å endre hvert partikkel sine attributte

Prøv selv først! Dersom du ikke får det til kan du benytte

Hint

For-løkke

- ☐ En `for`-løkke som skal gå gjennom en liste vil se slik ut:

```
for(var i = 0; i<listeNavn.length; i++){  
    //kode  
    element = listeNavn[i] // element blir nå det i-te ele  
il lengden av listen.  
}
```

Oppbygning av koden

For at du skal kunne bygge opp koden slik at partiklene oppfører seg som vi ønsker, må du vite hvor vi putter koden vår.

- ☐ All endring på partikkel-objektet bør skje i `for`-løkken. På den måten blir animasjonen blir finere.
- ☐ Når man bør legge elementer i `partikkel`-lista bør du eksperimentere litt med når du bruker `clearRect`.
- ☐ Du bør også eksperimentere litt med når du bruker `clearRect` legger den i eller utenfor `for`-løkken?

