

Om oppgaven

Denne oppgaven viser deg hvordan du kan lage et spill med JavaScript og dele det med vennene dine. Spillet kalles *Trykkomania* fordi det handler om å trykke på en ball flest mulig ganger før tiden renner ut.

Oppgaven henter ideer fra utvikling av web-applikasjoner med bibliotek som [React](#) og [Mithril](#), der elementer i spillet lages som individuelle komponenter. Komponentene ligner på objektorientert programmering, men bruker teknikken "closures" (funksjoner som husker konteksten de ble laget i).

✓ Oppgaven passer til:

Fag: Matematikk, Programmering, Informasjonsteknologi 2

Trinn: 4. trinn - VG3

Tema: JavaScript, web, variabler, closures, objektsorientering, objekter, funksjoner, HTML, CSS

Nivå: Nybegynner

Tidsbruk: Dobbeltime eller mer.

Kompetansemål

- ☐ **Matematikk, 4. trinn:** bruke matematiske symboler og uttryksmåter for å uttrykke matematiske sammenhenger i oppgaveløsning
- ☐ **Matematikk, 4. trinn:** tegne, bygge, utforske og beskrive geometriske figurer og modeller i praktiske sammenhenger, medregnet teknologi og design
- ☐ **Matematikk, 4. trinn:** lese av, plassere og beskrive posisjoner i rutenett, på kart og i koordinatsystemer, både med og uten digitale verktøy
- ☐ **Matematikk, 7. trinn:** beskrive plassering og flytting i rutenett, på kart og i koordinatsystem, med og uten digitale hjelpemidler, og bruke koordinater til å beregne avstander parallelt med aksene i et koordinatsystem
- ☐ **Programmering, 10. trinn:** dokumentere og forklare programkode gjennom å skrive hensiktsmessige kommentarer og ved å presentere egen og andres kode
- ☐ **Programmering, 10. trinn:** bruke grunnleggende prinsipper i programmering, slik som løkker, tester, variabler, funksjoner og enkel brukerinteraksjon
- ☐ **Programmering, 10. trinn:** overføre løsninger til nye problemer ved å generalisere og tilpasse eksisterende programkode og algoritmer.
- ☐ **Informasjonsteknologi 2, VG3:** utvikle og sette sammen delprogrammer
- ☐ **Informasjonsteknologi 2, VG3:** definere variabler og velge hensiktsmessige datatyper
- ☐ **Informasjonsteknologi 2, VG3:** tilordne uttrykk til variabler
- ☐ **Informasjonsteknologi 2, VG3:** lage egne og bruke egne og andres funksjoner eller metoder med parametere

- ☐ **Informasjonsteknologi 2, VG3:** planlegge og utvikle multimedieapplikasjoner ved å kombinere egne og andres multimedieelementer av typene tekst, bilde, lyd, video og animasjoner
- ☐ **Informasjonsteknologi 2, VG3:** bruke programmeringsspråk i multimedieapplikasjoner
- ☐ **Informasjonsteknologi 2, VG3:** programmere med valg og gjentakelser

Forslag til læringsmål

- ☐ Eleven kan bruke enkle matematiske uttryksmåter for å øke eller minke variabler i JavaScript.
- ☐ Eleven kan bruke JavaScript til å tegne en sirkel.
- ☐ Eleven kan plassere et element i på en nettside ved hjelp av koordinater på x- og y-aksen.
- ☐ Eleven kan skrive kommentarer til sin egen kode i JavaScript.
- ☐ Eleven kan bruke variabler, løkker og funksjoner til å manipulere elementer i JavaScript.
- ☐ Eleven kan videreutvikle sitt ferdige produkt ved hjelp av egenprodusert JavaScript-kode.

Forslag til vurderingskriterier

- ☐ Eleven oppnår middels måloppnåelse ved å fullføre oppgaven.
- ☐ Eleven oppnår høy måloppnåelse ved å videreutvikle egen kode basert på oppgaven.

Forutsetninger og utstyr

- ☐ **Forutsetninger:** Oppgaven er *kun* javascript, men det lønner seg å ha kjennskap til HTML og CSS.
- ☐ **Utstyr:** Datamaskin med internett.

Konsepter brukt i oppgaven

- ☐ Variabler
- ☐ Objekter
- ☐ Funksjoner
- ☐ Closures, funksjoner som husker konteksten de ble laget i.
- ☐ HTML-elementer via javascript
- ☐ CSS-stil via javascript
- ☐ onclick
- ☐ setInterval

Løsning

Her er en full løsning av oppgaven.

Fremgangsmåte

Her kommer tips, erfaring og utfordringer til de ulike stegene i den faktiske oppgaven. [Klikk her for å se oppgaveteksten.](#)

Generelt i oppgaven

- ☐ Elevene bør ha god og oversiktlig kode sånn at det er enkelt å finne feil. Dette oppnås ved å kommentere koden, samt bruke inntrykk og mellomrom mellom funksjoner og annen kode.
- ☐ Elevene må passe på at variabelnavn og tegnsetting er riktig.

Steg 2: Lage en ball

- ☐ Elevene kjenner kanskje igjen CSS-elementer når de skal lage funksjonen `Ball()`. Her vises det at HTML og CSS kan programmeres gjennom JavaScript.

Steg 3: Flytte ballen

- ☐ Elevene kan lure på hvor de skal legge til koden i dette steget, den skal legges til i funksjonen `Ball()` fordi `el` er en lokal variabel til funksjonen `Ball()`.

Steg 4: Flytte ballen med en funksjon

- ☐ Elevene kan være forvirret hva `x` og `y` er i denne oppgaven så her er det viktig å poengtere at dette er verdier som blir sendt inn senere i programmet, som vi ser rett før *Steg 5*.

Steg 5: Velg en tilfeldig plassering

- ☐ Her ser vi at vi kan legge til *strenger* bak tall som er blitt regnet ut: `Math.random() * 100 + '%'`. Dette kan være svært nyttig for elevene å vite i senere oppgaver.
- ☐ Elever kan lure på hva `Math.random()` er. Og ved å si `Math.random()` så kaller vi på et bibliotek (*Math*), altså en innebygget JavaScript-fil, som inneholder funksjonen `random()`.

Steg 7: Poeng

- ☐ Elevene må kopiere koden akkurat som den står oppført, hvis ikke blir det fort feil.

Steg 8: Begrense tiden

- ☐ Elevene må kopiere koden akkurat som den står oppført, hvis ikke blir det fort feil.

Steg 9: Omstarte spillet

- ☐ Elevene må kopiere koden akkurat som den står oppført, hvis ikke blir det fort feil.

Variasjoner

- ☐ *Vi har dessverre ikke noen variasjoner tilknyttet denne oppgaven enda.*

Lisens: [CC BY-SA 4.0](#)