

## 1.1

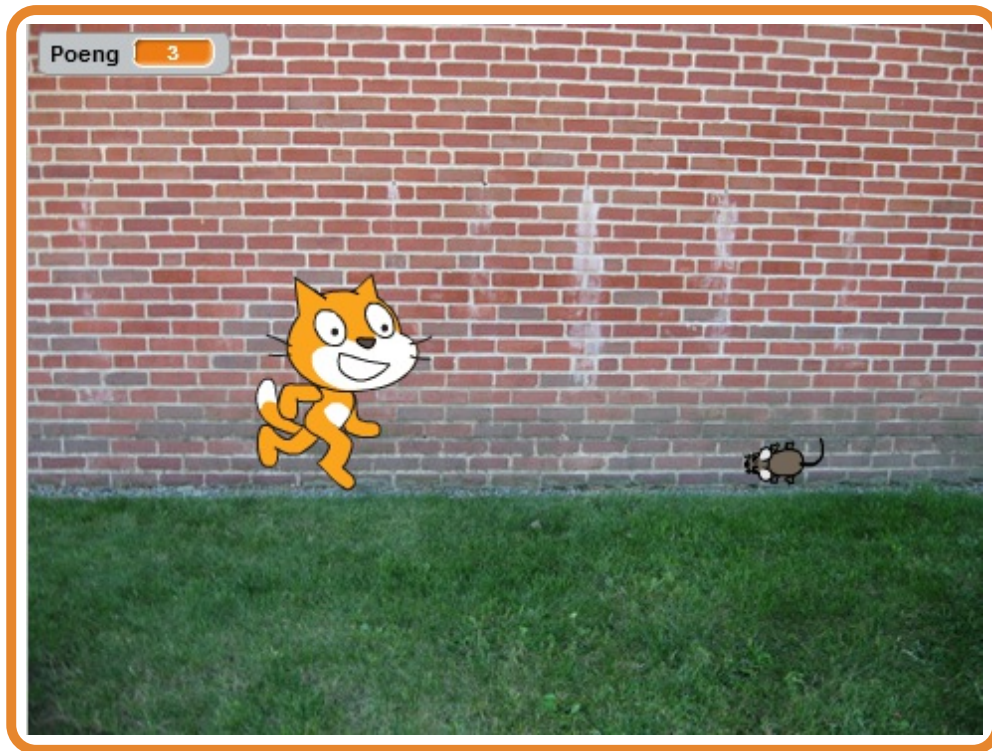
## Felix og Herbert



Dette kurset er utviklet av [Code Club \(http://www.codeclub.org.uk/\)](http://www.codeclub.org.uk/) i England, og oversatt til norsk av [Kodeklubben \(http://www.kodeklubben.no/\)](http://www.kodeklubben.no/).

### Introduksjon

Vi skal lage et spill hvor katten Felix skal fange musa Herbert. Du styrer Herbert med musepekeren og skal prøve å unngå å bli tatt av Felix. Jo lenger du unngår ham jo flere poeng får du, men blir du tatt, går poengsummen din ned.



### Steg 1: Felix følger musepekeren

Vi ønsker at katten Felix skal følge etter musepekeren.



Sjekkliste




Start et nytt prosjekt.



Trykk på **i** i hjørnet av **Sprite1** og bytt navn på figuren til Felix.



Sørg for at Felix kun ser til høyre og venstre ved å sette rotasjonsmåte til **↔**.

Klikk på scenen ved siden av Felix i vinduet for figurer. Velg fanen **Bakgrunner** og trykk på  for å importere en ferdig bakgrunn. Velg bakgrunnen Utendørs/brick wall2.



Velg **Skript**-fanen og lag dette skriptet:



## Test prosjektet

Klikk på det grønne flagget.

- Følger Felix musepekeren?
- Ser det ut som han går når han beveger seg?
- Beveger han seg med riktig hastighet?



## Lagre prosjektet




Scratch lagrer alle prosjektene dine automatisk med jevne mellomrom. Det kan likevel være lurt å passe på at prosjektet faktisk er lagret innimellom.

- I filmenyen, velg Lagre nå.

## Steg 2: Felix jager Herbert

Nå ønsker vi at Felix skal jage musa Herbert i stedet for musepekeren.

## Sjekkliste

- ☐ Lag en ny figur ved å trykke på  og velg figuren Dyr/Mouse1.
- ☐ Bytt navn på figuren til Herbert og sørg for at også Herbert kun kan se til høyre og venstre, .
- ☐ Gjør Herbert mindre enn Felix ved å trykke på  (øverst mot midten av vinduet). Prøv seks klikk.
- ☐ Gi Herbert dette skriptet:



---

## Test prosjektet

Klikk på det grønne flagget.

- Flytter Herbert seg med musepekeren?
- Jager Felix Herbert?

## Steg 3: Felix sier når han har fanget Herbert

Vi vil at Felix skal vite når han har fanget Herbert og fortelle det til oss.

## Sjekkliste

- ☐ Endre skriptet til Felix til dette:



---

## Test prosjektet

Klikk på det grønne flagget.

- Sier Felix fra når han har fanget Herbert?

## Steg 4: Herbert blir et spøkelse når han fanges

I stedet for at Felix sier noe, vil vi at Herbert blir forvandlet til et spøkelse når han fanges.

### Sjekkliste

- ☐ Endre skriptet til Felix slik at det sender en melding og lager en lyd når han fanger Herbert



- ☐ Velg Herbert og gå til Drakter-fanen.
- ☐ Hent en ny drakt ved å trykke på 🦇 og velg Fantasi/ghost2-a
- ☐ Gjør drakten mindre ved å velge ⏏ og trykke seks ganger på spøkelsesdrakten.
- ☐ Endre navnene på Herberts drakter slik at musedrakten heter levende og spøkelsesdrakten heter død.
- ☐ Lag et nytt skript for Herbert for å gjøre ham om til et spøkelse. Ikke slett det gamle skriptet



---

## Test prosjektet

Klikk på det grønne flagget.

- Forvandles Herbert til et spøkelse når han fanges?
- Spiller Felix de riktige lydene til riktig tid?
- Står Felix stille lenge nok til at Herbert kommer seg unna?

## Steg 5: Telle poeng

La oss legge til en poengsum slik at vi kan se hvor flink man er til å holde Herbert i live. Vi begynner med poengsummen null og øker den med en for hvert sekund. Hvis Felix fanger Herbert, minker vi poengsummen med hundre.

## Sjekkliste

- ☐ På Skript-fanen under kategorien **Data**, lag en ny variabel. La den gjelde for alle figurer og kall den **Poeng**.



- ☐ Lag disse to skriptene på scenen:



---

## Test prosjektet

Klikk på det grønne flagget.

- Øker poengsummen med en hvert sekund?
- Går poengsummen ned med hundre når Herbert blir fanget?
- Hva skjer når Herbert fanges før du har hundre poeng?
- Går poengsummen tilbake til null når du starter spillet på nytt?

---

## Lagre prosjektet

Du er ferdig. Godt gjort. Nå kan du spille spillet!

Husk at du kan dele spillet med familie og venner ved å trykke **Legg ut** på menylinjen.

Dette kurset er utviklet av [Code Club \(http://www.codeclub.org.uk/\)](http://www.codeclub.org.uk/) i England, og oversatt til norsk av [Kodeklubben \(http://www.kodeklubben.no/\)](http://www.kodeklubben.no/).

## 1.2

## Spøkelsesjakten



Dette kurset er utviklet av [Code Club](http://www.codeclub.org.uk/) (<http://www.codeclub.org.uk/>) i England, og oversatt til norsk av [Kodeklubben](http://www.kodeklubben.no/) (<http://www.kodeklubben.no/>).

### Introduksjon

Dette prosjektet er inspirert av tivolispillet Whack-a-mole, hvor man slår muldvarper ned igjen i hullene sine. I vårt spill er det spøkelsene som blir borte når vi klikker på dem. Målet er å klikke bort flest mulig i løpet av 30 sekunder.



### Steg 1: Lag et flyvende spøkelse



#### Sjekkliste



Start et nytt Scratch-prosjekt.

Slett katterfiguren ved å høyreklikke på den og velge slett





Endre bakgrunnen til Natur/woods.

For å legge til et spøkelse klikker du på -knappen. Velg Fantasi/Ghost1-figuren.

Gi spøkelset navnet spøkelse1.

Du skal nå lage en variabel som styrer hvor fort spøkelset beveger seg. Denne kan vi senere bruke til å endre hastigheten når spillet er i gang.



Under Skript, trykk på **Data** og så Lag en variabel. Kall variabelen hastighet. Huk av der det står For denne figuren.



På scenen skal variabelen hete spøkelse1: hastighet. Hvis den bare heter hastighet, så slett den og legg den til en gang til.



Fjern avhukingen ved siden av variabelen, slik at den ikke vises på scenen:



Vi vil at spøkelset skal bevege seg når spillet starter. Det gjør vi ved å lage følgende skript



## Test Prosjektet

Klikk på det grønne flagget.

- Hvorfor sitter spøkelset fast når det treffer kanten på skjermen?




## Sjekkliste



For å unngå at spøkelset sitter fast i veggen må vi få det til å snu når det treffer den. Dette gjøres ved å legge til blokken **sprett tilbake ved kanten**. Skriptet ser da slik ut:



For å hindre at spøkelset snur seg opp-ned, klikk på -knappen i spøkelsets infoboks.



## Test prosjektet

Klikk på det grønne flagget.

### Ting å prøve



Endre hastighetsvariabelen, slik at spøkelset går raskere eller saktere.



Hvordan kan vi få spøkelset til å fly fortere jo lenger det flyr? (Dette er nokså vanskelig, så ikke vær bekymret hvis du ikke skjønner hvordan. Du vil få flere hint underveis.)

## Steg 2: Få spøkelset til å dukke opp og forsvinne

For å gjøre spillet morsommere vil vi få spøkelset til å dukke opp og forsvinne.



### Sjekkliste



Vi lager et nytt skript, som skal kjøre samtidig med skriptet som beveger på spøkelset. Det skriptet gjemmer bort spøkelset en tilfeldig periode og viser det deretter frem igjen i en tilfeldig periode. Dette skal skje om og om igjen, frem til spillet er slutt. Slik lager du skript



## Test prosjektet

Klikk på det grønne flagget.

- Flytter spøkelset seg fra side til side? Forsvinner det og dukker opp igjen helt tilfeldig? Da er alt riktig.

### Ting å prøve

- ☐ Prøv å endre tallene i koden der det står `tilfeldig tall fra () til ()`. Hva skjer hvis du velger veldig store eller små tall? (Dette gir deg kanskje et nytt hint for hvordan vi skal få spøkelset til å gå fortere jo lenger man spiller.)

## Steg 3: Tryll bort spøkelset med et klikk!

For å gjøre dette til et ordentlig spill må vi gi spilleren noe å gjøre - for eksempel å klikke bort spøkelset. Når det skjer vil vi også at det skal komme en stilig tryllelyd!

### Sjekkliste

- ☐ Fra **Lyd**-boksen henter du lyden Elektronisk/fairydust.
- ☐ Så lager du skriptet som får spøkelset til å forsvinne når det blir klikket på:



---

### Test prosjektet

Klikk på det grønne flagget.

- Forsvinner spøkelset med en tryllelyd når du klikker på det?

### Ting å prøve

- ☐ Spør hjelperen din om du kan ta opp en egen lyd. Denne kan brukes istedenfor tryllelyden.

## Steg 4: Legg til tid og poeng

Vi har et spøkelse å trylle bort, så nå vil vi ha poeng for å gjøre det! Vi vil også ha en tidsfrist, slik at det er om å gjøre å få flest mulig poeng på denne tiden. Vi løser begge deler ved å bruke variabler.

### Sjekkliste

- ☐ Lag en ny variabel som heter **poeng**. Denne skal gjelde for alle figurer. Legg til en ny blokk som gjør at poengvariabelen økes med 1 poeng for hver gang spilleren klikker på spøkels



- ☐ Klikk på Scene og lag en ny variabel som heter tid. La variabelen vises på skjermen.
- ☐ Lag et nytt skript som setter tid-variabelen til 30 og poeng-variabelen til 0 når det grønne flagget klikkes.
- ☐ Bruk så en **gjenta til** -blokk for å vente i 1 sekund og deretter redusere tiden med 1 sekund. Denne skal kjøre fram til tiden er ute. Til slutt stopper du hele spillet med en **stopp alle** -blokk.



---

### Test prosjektet

Klikk på det grønne flagget.

## Ting å prøve



Hvordan kan du få spøkelset til å gå fortere når spillet er i gang?



### Lagre prosjektet

- Bra jobba!. Nå er du egentlig ferdig med spillet, men prøv deg gjerne på neste steg også.

### En ekstra utfordring: Flere spøkelser!

Hvis ett spøkelse er bra, må vel flere være enda bedre! La oss ha tre spøkelser flyvende rundt!



Lag flere spøkelser ved å høyreklikke på det du allerede har, og så kopiere denne.



La spøkelsene få ulik størrelse. Dette gjør du ved først å klikke på **Drakter**, og så én av knappene under. Den til venstre forstørrer figuren og den til høyre



forminsker:



Du kan også endre spøkelsenes flyvefart. Dette gjøres i hastighet-variabelen i det øverste skriptet for hver enkelt figur.



Til slutt kan du spre spøkelsene litt bedre ut på scenen. Dette gjør du ved å klikke og dra figurene rundt i selve skjermbildet.

## Test prosjektet

Klikk på det grønne flagget.

- Har du nå tre spøkelser som flyr fra side til side? Som plutselig forsvinner og dukker opp igjen? Forsvinner når du klikker på dem? Gratulerer! Da har du gjort alt riktig!

## Ting å prøve

- ☐ Hvor mange spøkelser synes du spillet fungerer best med? Legg til flere og prøv!
- ☐ Klarer du å få spøkelsene til å se forskjellige ut? Klikk på **Drakter** og prøv deg frem. Du kan også velge noen av blokkene under **Utseende**.
- ☐ Kan du få spøkelsene til å bli verdt forskjellige antall poeng? Hva med å få den minste og raskeste til å gi 10 poeng?



## Lagre prosjektet

Bra jobba! Nå er du ferdig, og det er på tide med litt seriøs spilling. Husk også at du kan dele spillet med vennene dine. Det gjør du ved å klikke på **Legg ut** i toppmenyen.

Dette kurset er utviklet av [Code Club \(http://www.codeclub.org.uk/\)](http://www.codeclub.org.uk/) i England, og oversatt til norsk av [Kodeklubben \(http://www.kodeklubben.no/\)](http://www.kodeklubben.no/).

## 1.3

## Fyrverkeri



Dette kurset er utviklet av [Code Club \(http://www.codeclub.org.uk/\)](http://www.codeclub.org.uk/) i England, og oversatt til norsk av [Kodeklubben \(http://www.kodeklubben.no/\)](http://www.kodeklubben.no/).

### Introduksjon

I dette prosjektet skal vi skyte opp fyrverkeri over en by.



### Steg 1: Lag en rakett som flyr mot muspekeren


Vi starter med å importere forskjellige bilder vi skal bruke i spillet



#### Sjekkliste



Lag et nytt Scratch-prosjekt. Fjern katten ved å høyreklikke på den og velge slett

- ☐ Bytt bakgrunnsbilde til utendørs/city-with-water
- ☐ Klikk på Ny figur: Last opp figur fra fil, , for å legge til en rakett-figur i prosjektet, for eksempel ressurser/rocket.png
- ☐ Vi vil at raketten skal skjules når du klikker på det grønne flagget.



Nå vil vi gjerne at raketten skal bevege seg mot muspekeren når du klikker med musa.

- ☐ Legg til en kloss **når mellomrom trykkes**. Deretter legger vi to klosser som gjør raketten synlig og lar den bevege seg mot muspekeren.



---

## Test prosjektet

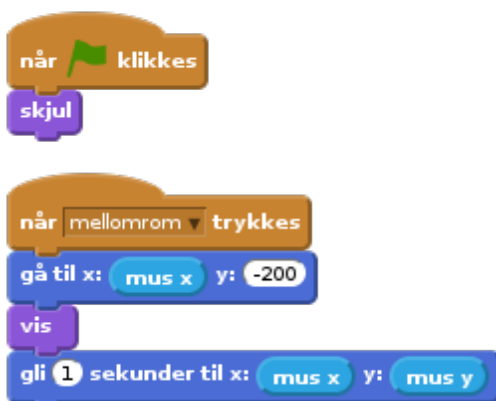
Klikk på det grønne flagget, plasser muspekeren over scenen og trykk mellomromstasten.

- Får du opp raketten som beveger seg mot muspekeren?
- Hva skjer hvis du flytter på muspekeren og trykker mellomromstasten igjen?

## Sjekkliste

Fyrverkeri pleier ikke å fly fra side til side, så du bør gjøre det slik at raketten alltid flyr mot muspekeren fra bunnen av skjermen.

- ☐ Før du fyrer opp raketten: bruk klossen **gå til** for å få raketten til å flytte seg til bunnen av skjermen, men bli værende samme sted horisontalt.





---

## Test prosjektet

Klikk på det grønne flagget, plasser musa over scenen og trykk mellomromstasten.

- Flyr raketten mot muspekeren fra bunnen av skjermen?
- Hva skjer hvis du flytter på musa og trykker mellomromstasten igjen?

## Sjekkliste

- ☐ Endelig, prøv å få til det samme ved å bruke musknappen istedenfor mellomromstasten. For å gjøre dette kan vi pakke skriptet vårt inn i en **for alltid - hvis museknappen er nede**-blokk.
- ☐ Bytt så klossen **når mellomrom trykkes** til **når grønt flagg klikkes** og sist men ikke minst sørg for at raketten er skjult når alt starter opp.



---

## Test prosjektet

Klikk på det grønne flagget


- Trykk musknappen over scenen. Klikk igjen et annet sted.

### Utfordringer

- ☐ Prøv å få noen raketter til å bevege seg litt saktere eller fortere enn andre.
- ☐ Prøv å endre måten raketten snur seg mot muspekeren på: få den til å bue seg litt.

## Steg 2: Få raketten til å eksplodere

### ✓ Sjekkliste

- ☐ Første steg for å få raketten til å eksplodere er å spille av en bang-lyd ( [ressurser/bang.wav](#) ) før den begynner å bevege seg og så gjemme seg når den når muspekeren. For å importere en lyd gå til fanen **Lyder** og klikk på Last opp lyd fra fil, .



- ☐ Neste steg er å få raketten til å sende en melding til resten av spillet når den eksploderer. Den skal lytte etter meldingen senere. Lag en ny melding som heter **eksploder**.



Klikk på det grønne flagget.

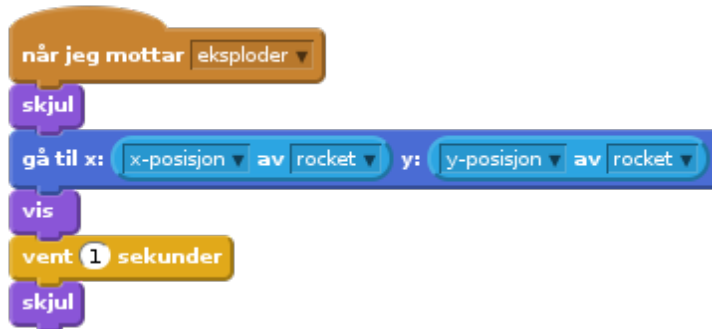
- Sørg for at raketten spiller av lyden og gjemmer seg når den når muspekeren.

## Sjekkliste



Last opp en ny figur fra fil (ressurser/firework1.png).

Når denne figuren får meldingen **eksploder** passer vi på at den er gjemt, flytter den til raketten ved bruk av klossen **gå til**, viser den og skjuler den igjen 1 sekund senere.



## Test prosjektet

Fyr av en rakett til.

- Blir den erstattet med et eksplosjonsbilde når den eksploderer?
- Hva skjer hvis du holder musknappen nede mens du beveger på musa? (Ikke bekymre deg, vi skal fikse det senere).


## Steg 3: Gjør hver eksplosjon unik

## Sjekkliste



Nå kan vi gjøre hver eksplosjon enda mer unik ved å bruke klossen **sett fargeeffekt** og vel en tilfeldig farge før eksplosjonen vises.



- ☐ Legg til forskjellige bilder av eksplosjoner som drakter ved å velge Drakter-fanen til firework1. Ved å klikke Last opp drakt fra fil, , kan du legge til firework2.png, firework3.png og firework4.png fra ressurser.
- ☐ Klarer du å få eksplosjonene til å bruke forskjellige drakter? (Hint: Du kan for eksempel bruke neste drakt et passende sted i skriptet til Firework1.)



## Test prosjektet

Klikk på det grønne flagget.

- Har hver eksplosjon en forskjellig farge?
- Har hver rakett et forskjellig bilde av eksplosjonen?



## Sjekkliste

- ☐ Endelig, gjør eksplosjonen større etter at raketten eksploderte! I stedet for å vente i 1 sekund sett størrelsen til figuren til 5% før den vises, og så når den blir synlig øk størrelsen med 2 femti ganger ved bruk av klossen **gjenta**.





## Test prosjektet

Klikk på det grønne flagget.

- Har bildet av eksplosjonen spredt ut fra midten av raketten?
- Vokser eksplosjonen gradvis?

### Utfordringer

Prøv å gjøre hver eksplosjon unik: endre størrelsen og veksthastigheten for eksplosjonen.

## Steg 4: Fikse "send melding"-feilen

Husker du at vi tidligere hadde et problem med å holde museknappen nede?

Dette problemet oppstår fordi når raketten sender sin melding om eksplosjonen, gjentar den hvisløykka med en gang. Dermed blir det sendt en eksplosjonsmelding før den forrige er ferdig med animasjonen.

I programmeringsverden kaller vi denne type problemer for bugs, fordi man i gamle dager (da datamaskiner var mye større) kunne ha problemer med at innsekter ble fanget inne i datamaskinene og ødela programmer.



### Sjekkliste



For å fikse dette kan du erstatte klossen `send melding` med `send melding og vent`. Da vil ikke løkken gjentas før den forrige eksplosjonen er ferdig.



---

## Test prosjektet

Klikk på det grønne flagget, hold museknappen nede og beveg litt på musa.

- Får du eksplosjonen til å dukke opp på riktig sted og til riktig tid?

---

## Lagre prosjektet

Gratulerer, du er ferdig! Nå kan du kose deg med spillet!

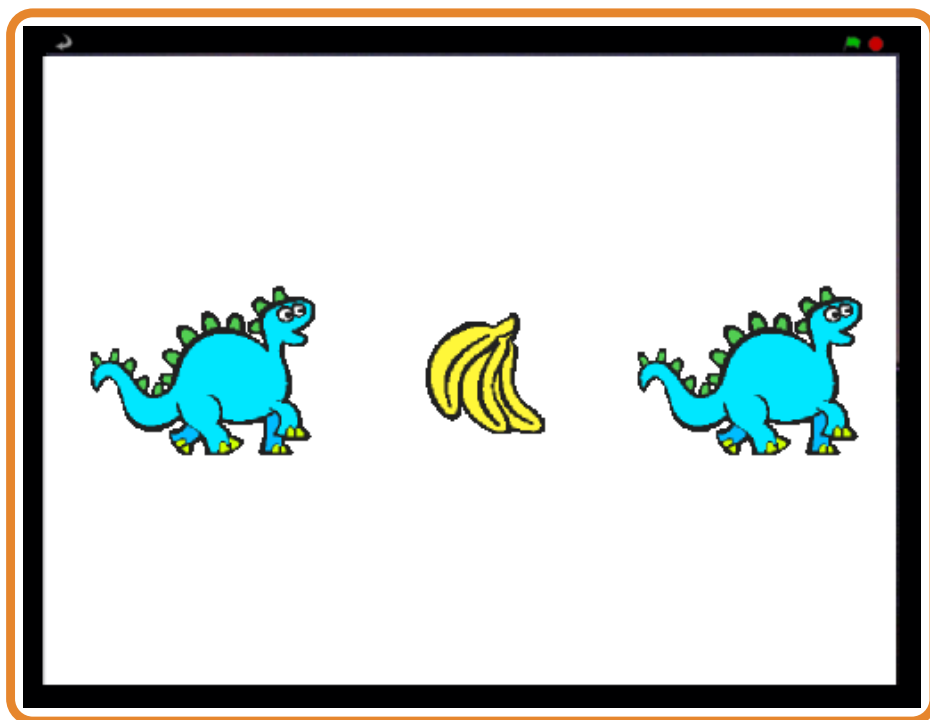
Ikke glem å dele spillet ditt med alle dine venner og familien! Klikk på **Legg ut** i menylinjen.

Dette kurset er utviklet av [Code Club \(http://www.codeclub.org.uk/\)](http://www.codeclub.org.uk/) i England, og oversatt til norsk av [Kodeklubben \(http://www.kodeklubben.no/\)](http://www.kodeklubben.no/).

Dette kurset er utviklet av [Code Club \(http://www.codeclub.org.uk/\)](http://www.codeclub.org.uk/) i England, og oversatt til norsk av [Kodeklubben \(http://www.kodeklubben.no/\)](http://www.kodeklubben.no/).

## Introduksjon

Dette er et spill med tre figurer som endrer utseende. Spillerens oppgave er å stoppe figurene én etter én, slik at alle blir like.



## Steg 1: Lag en figur som bytter drakt

La oss importere de bildene vi trenger for spillet



### Sjekkliste



Start et nytt Scratch-prosjekt. Slett katten ved å høyreklikke og velge slett.

Importer en ny figur, . Velg den figuren du vil.

Gå til **Drakter**, og importer to ekstra drakter fra biblioteket, så figuren til sammen har tre drakter.

## Steg 2: Få figuren til å rullere

Nå som figuren har noen drakter, ønsker vi at den skal veksle mellom dem.

### ✓ Sjekkliste

- ☐ Klikk på Skript-fanen,
- ☐ Velg **Hendelser** og hent en **Når grønt flagg klikkes**-blokk. Denne vil kjøre hver gang du klikker på det grønne flagget over scenen.
- ☐ Hent en **for alltid**-blokk fra **Styring**-kategorien og legg den under den andre blokken slik at den fester seg til den.
- ☐ Klikk på det grønne flagget. Du ser at du får en hvit linje rundt skriptet. Det betyr at skriptet kjører.
- ☐ Velg nå **Utseende** og dra en **neste drakt**-blokk inn i **for alltid**-blokken.
- ☐ Hvordan kan vi få draktbyttet til å gå saktere? Velg **Styring** og dra inn en **vent (1) sekunder**-blokk.
- ☐ Tilpass tiden til figuren endrer drakt i et hurtigere tempo (0.5 sekunder ser bra ut). Hva tror du ville skjedd om vi ikke hadde med **vent**-blokken?



### Test Prosjektet

Klikk det grønne flagget.

- Endrer figuren drakt i et fornuftig tempo?

#### Ting å prøve

Tilpass tiden i **vent**-blokken. Hvilke tall gjør spillet for vanskelig eller for lett?

## Steg 3: Frys ruletten!

### ✓ Sjekkliste



Bra! Vi kan få draktene til å bytte i det uendelige, men hvordan får vi de til å stoppe når vi klikker på det? En måte å gjøre dette på er ved å bruke en variabel som setter statusen til figuren. Dette vil også være praktisk senere...

- ☐ 1. Klikk på **Data** og lag en ny variabel. Kall den **stoppet** og huk av for For denne figuren. Fjern avhukingen foran variabelen slik at den ikke vises på scenen.
- ☐ 2. På starten av spillet vil ikke figuren ha blitt klikket så da setter vi variabelen til 0.



- ☐ 3. Nå vil vi sette variabelen **stoppet** til 1 når noen trykker på figuren.



- ☐ 4. Til slutt må vi få figuren til å slutte å forandre drakt når variabelen **stoppet** blir "1". Legg til en **hvis-løkke** og bruk en er lik **=** operator blokk (se under Operatorer) for å sjekke om **stoppet** fremdeles er "0".



---

## Test prosjektet

Klikk på det grønne flagget, vent et øyeblikk og klikk så på figuren.

Endres drakten før du klikker på den?

Stopper den når du klikker på den?

Start skriptet igjen.

Stopper den når du setter pekeren over den uten å klikke?

Stopper den når du klikker andre steder på scenen eller andre steder i Scratch?



Lagre prosjektet

## Steg 4: Lag de andre figurene

Nå trenger vi to figurer til for å gjøre spillet komplett!



### Sjekkliste



1. Dupliser figuren din (Figuren) ved å høyreklikke og velg lag en kopi.



2. Lag en kopi til slik at du har tre figurer på skjermen. Vi har kalt våre figurer Figur1, Figur2 og Figur3.



3. Flytt figurene slik at de er en linje. Gjør dem mindre med krympeknappen hvis det trengs med



### Test prosjektet

Klikk på det grønne flagget. alle figurene skal nå forandre seg. Prøv å stoppe dem, én for én!



Lagre prosjektet

## Steg 5: Start hver figur med en tilfeldig drakt

La oss få figurene til å skifte til en tilfeldig drakt når det grønne flagget klikkes.

Når du starter spillet rett vil du se at alle figurene skifter drakt samtidig. Spillet vil bli mer gøy (og vanskeligere) dersom de endres litt mer uforutsigbart.



### Sjekkliste



1. Hvis du ser under drakter-fanen til en figur vil du se at hver drakt har et nummer. Du kan spesifisere hvilken drakt figuren skal ha ved å bruke navnet eller nummeret.



2. For å få figuren til å starte med en tilfeldig drakt, la oss legge til en bytt drakt til blokk med tilfeldig tall fra (1) til (3) for å velge draktnummer.



3. Vi kan også bruke samme blokken i for alltid løkka slik at figuren skifter til en ulik drakt hver gang den forandres.



- ☐ 4. Gjør det samme for hver av figurene.

## Test prosjektet

Klikk på det grønne flagget. Alle figurene burde skifte drakter uforutsigbart.

Hvordan må vi forandre skriptet dersom vi legger til en annen drakt?

## Lagre prosjektet

### Ting å prøve

Gjør spillet vanskeligere!

Se om du klarer å endre vanskelighetsgraden på et eller annet vis. Å få draktene til å rullere raskere er enkelt. Prøv å gjøre noe litt mer oppfinnsomt. Noen muligheter du kan tenke på er:

- ☐ Endre antall drakter hver figur har.
- ☐ Gi noen av figurene helt forskjellige drakter.
- ☐ Endre tid mellom draktbytte.

Lek og kom opp meg egne idéer!

Hver gang du endrer noe, tenk på om det vil gjøre spillet lettere eller vanskeligere. Er spillet for lett eller for vanskelig? Hvordan kan du justere det slik at det blir akkurat passe?

## Steg 6: Vis en melding når spillet er over

La oss vise en "Game Over" melding når spillet er over

## Sjekkliste

La oss først hente en ny bakgrunn som vi viser når spillet er over.

1. Klikk på scenen og så **Bakgrunnen** fanen. Forandre navnet til den eksisterende bakgrunnen til "GameOn".
2. Dupliser bakgrunne og legg til en tekst som sier "Game over". Du kan forandre størrelsen på teksten ved å klikke på den og dra i hjørnene. Kall bakgrunnen "GameOver".
3. Klikk på **Skript** fanen for bakgrunnen og sett "GameOn" bakgrunne til å vises når spillet starter.
4. Hvordan kan vi se at alle figurene har stoppet? Husk at vi brukte **stoppet** variabelen til å sjekke om figurene hadde blitt klikket på. La oss sjekke **stoppet** variabelen for Figur3 figuren (din figur nr.3 kan ha annet navn) for å se om spillet er over. Velg den tredje frukten og bruk en **x-posisjon av Figur3** blokk i **Sansing**, men bytt ut x-posisjon med **stoppet**.

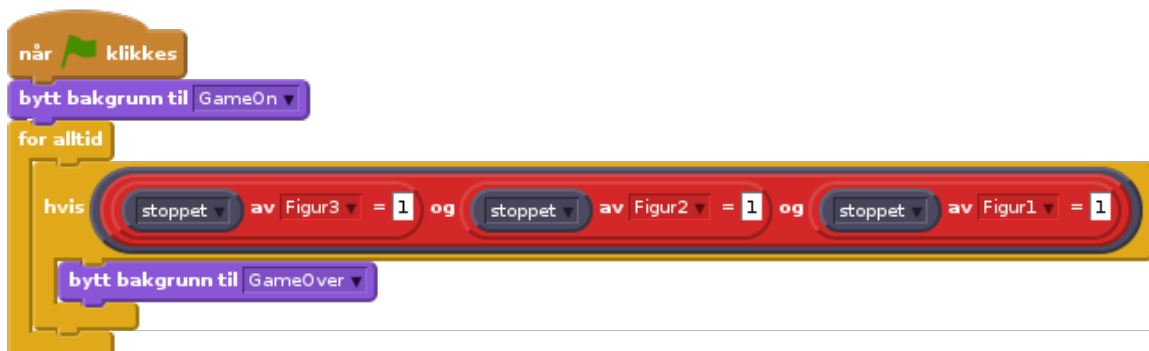


## Test prosjektet

Klikk på det grønne flagget. Vises "Game Over" meldingen når du trykker på Figur3?

Hva skjer dersom du stoppet Figur3 før du har klikket begge den andre fruktene? La oss forandre skriptet slik at det vil funke uansett hvilken rekkefølge figurene stoppes i.

5. For å sjekke om alle tre fruktene har satt sine **stoppet** variabler til 1, kan vi bruke **og** { . blockgreen } operatoren. Dette er en komplisert blokk som kan være litt trøblette å lage, så forsøk å gjøre ett steg av gangen.



## Test prosjektet

Klikk på det grønne flagget. Vises "Game Over" meldingen når alle tre fruktene er stoppet uansett hvilken rekkefølge du klikket på de?



## Lagre prosjektet

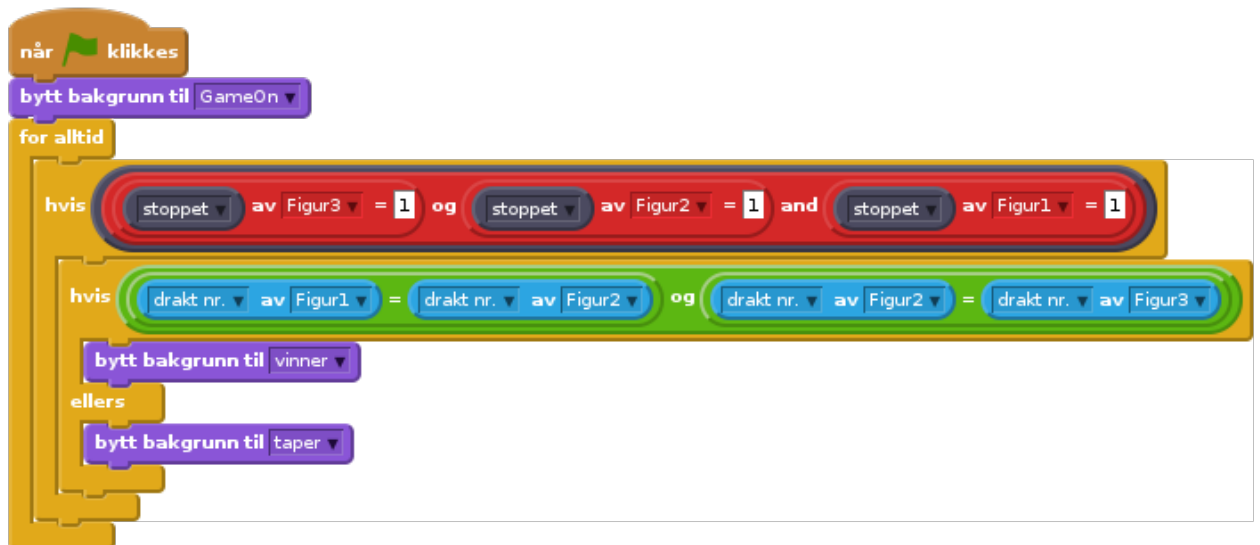
# Steg 7. Fortell spilleren om de vant eller tapte.

Målet med spillet er å klikke på figurene slik at de stopper når de viser samme drakten. Det ville være praktisk å vise en melding som forteller deg om du vant eller tapte.



## Sjekkliste

- ☐ 1. Vi har tidligere skrevet kode som sjekker om spillet var over, så alt vi trenger å gjøre er å sjekke om spilleren har vunnet. Gå tilbake til bakgrunnene og legg til mer tekst til GameOver bakgrunne slik at den også viser "Du vant!". Skift navn til "vinner".
- ☐ 2. Kopier bakgrunne og legg til en tekst hvor det står "Du tapte". Gi bakgrunnen navnet "taper".
- ☐ 3. Nå trenger vi kode for å velge hvilken bakgrunn vi skal vise når spillet er over. Vi kan bruke en **hvis...ellers** blokk for å se om brukeren har vunnet eller tapt ved å sammenligne **drakt nr.** (drakt nummer). Vi bruker en blokk som ligner på **x-posisjon av Figur** blokken vi brukte tidligere. Denne gangen, istedet for å se på **stoppet** variabelen, skal vi sjekke **drakt nr.** og se om Figur1 har samme drakt som Figur2 og om Figur2 har samme drakt som Figur3.



## Test prosjektet

Klikk på det grønne flagget. Vises den riktige meldingen når spillet er over? Hva skjer hvis draktnummerene ikke er like?



## Lagre prosjektet

Veldig bra! Du har nå fullført spillet, men det er fremdeles ting du kan gjøre med spillet ditt. Prøv deg på disse utfordringene!

## Utfordring: Gjør spillet enklere og vanskeligere med tiden

Alle er ikke like flinke til spillet. Hvordan kan du la vanskelighetsgraden avhenge av spilleren?

En måte å gjøre dette på er å endre hastigheten draktene forandres på. Du kan bruke en variabel kalt **forsinkelse**, for å gi varigheten til hver figurs venteblokk. Hvis spilleren vinner runden kan forsinkelsen reduseres litt (for å gjøre spillet vanskeligere). Hvis spilleren taper runden kan man øke forsinkelsen litt for å gjøre spillet lettere.

Du må sikkert vurdere å bruke en annen måte å starte spillet på istedet for å bruke **når grønt flagg klikkes**. Deretter kan du lagre verdiene i variabler som huskes mellom rundene.



### Lagre prosjektet

Godt gjort, du er ferdig! Nå kan du nyte spillet ditt! Ikke glem at du kan dele spillet med alle vennene og familien din ved å klikke på Legg ut i topp-menyen!

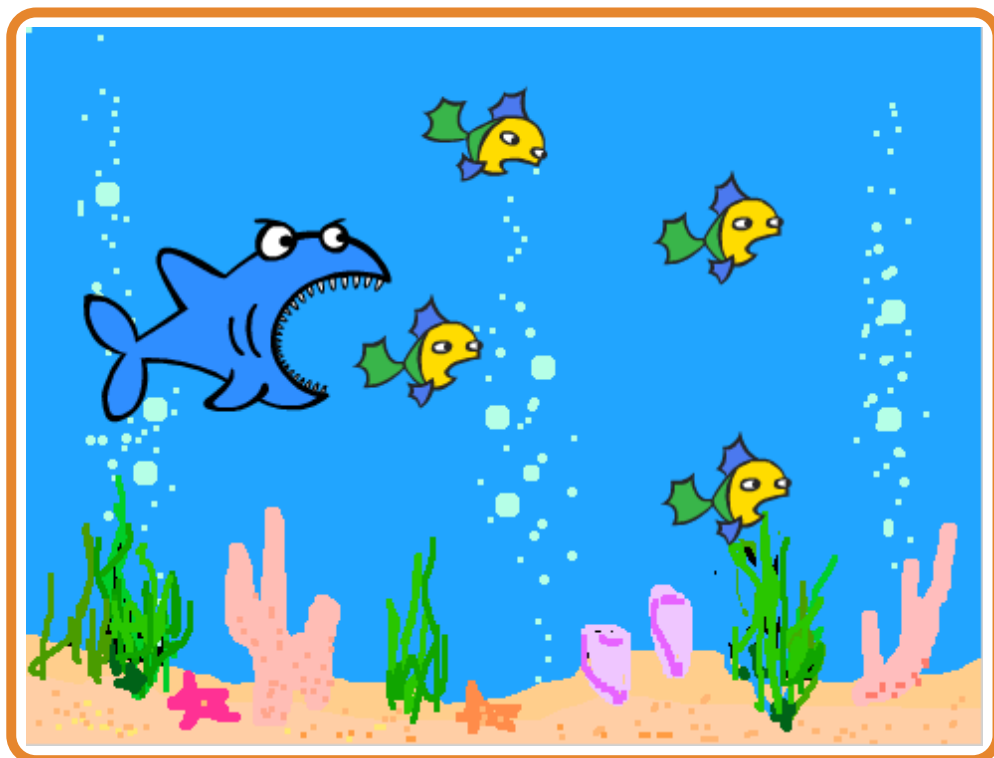
Dette kurset er utviklet av [Code Club \(http://www.codeclub.org.uk/\)](http://www.codeclub.org.uk/) i England, og oversatt til norsk av [Kodeklubben \(http://www.kodeklubben.no/\)](http://www.kodeklubben.no/).

Dette kurset er utviklet av [Code Club \(http://www.codeclub.org.uk/\)](http://www.codeclub.org.uk/) i England, og oversatt til norsk av [Kodeklubben \(http://www.kodeklubben.no/\)](http://www.kodeklubben.no/).

## JafseFisk

### Introduksjon

Vi skal nå lage et JafseFisk-spill! Målet i spillet er å hjelpe JafseFisk med å spise alle byttedyrene som svømmer rundt i havet.



skjerm bilde

## Steg 1: JafseFisk følger musepekeren

Først skal vi lage JafseFisk som svømmer rundt i havet!

## Sjekkliste

- ☐ Start et nytt Scratch prosjekt.
- ☐ Riktig bakgrunn får du ved å velge Scene og så Bakgrunn-fanen. Importer bakgrunnen Natur/underwater3 ved å velge Velg en ferdig bakgrunn. Slett så den andre bakgrunnen backdrop1.
- ☐ Endre Sprite1's navn til JafseFisk ved å trykke på det blå 'i' symbolet.
- ☐ Gi figuren en haidrakt ved å velge Velg drakt fra biblioteket. Velg drakt Dyr/shark-b.png. Kall drakten åpen munn. Slett så figurens andre drakt (costume1 og costume2).
- ☐ Klikk på det blå 'i' symbolet igjen, og pass på at figuren bare kan bevege seg fra side til side ved å velge rotasjonsmåte <-->.
- ☐ Få fisken til å følge musepekeren rundt i sjøen ved å lage dette skriptet:



---

## Test Prosjektet

Klikk det grønne flagget. Flytt musepekeren rundt i sjøen. Følger fisken med? Hva skjer hvis du ikke flytter musepekeren, og fisken når den igjen? Hvordan ser den ut? Hvorfor gjør den dette?

- Du kan stoppe JafseFisks maniske flipping hvis du sørger for at den bare flytter seg når den ikke er for nær musepekeren (avstand til blokken ligger under Sansning paletten).





## Ting å prøve

Hvis du vil kan du forandre tallene i skriptet, og se hvordan det forandrer bevegelsene? Sett avstandsgrensen til et stort tall (f.eks. 100), eller et lite tall (f.eks. 1). Sett antall steg fisken flytter seg til et stort tall (f.eks. 20) eller et lite tall (f.eks. 1, eller 0).

## Steg 2: Legg til byttedyr

Det er på tide å gi JafseFisk noe å spise!

### Sjekkliste

- ☐ Lag en ny figur fra biblioteket ved å bruke Dyr/Fish2.
- ☐ Gjør figuren mindre med krympeknappen som ligger over den røde stopp-knappen.
- ☐ Få byttedyrene til å bevege seg i tilfeldige retninger. Først skal vi la dem bevege seg litt framover, og så snu en tilfeldig valgt vinkel med eller mot klokka, og deretter gjenta.



---

### Test prosjektet

Klikk på det grønne flagget og se hvordan byttedyret svømmer rundt. Svømmer det slik du forventet? Ser bevegelsene naturlige ut?

For øyeblikket samspiller ikke JafseFisk og byttedyret med hverandre. Det skal vi gjøre noe med i neste steg.

---

### Lagre prosjektet

## Ting å prøve

- ☐ Prøv å forandre tallene for steg og tilfeldig tall. Hvordan forandrer det byttedyrenes bevegelser?
- ☐ Hva gjør sprett tilbake ved kanten blokken? Fjern blokken og se hva som skjer.

## Steg 3: JafseFisk spiser byttet

Nå skal vi la JafseFisk spise byttet! Når den har fanget byttet i munnen skal to ting skje: \* Den må lukke munnen og lage en gomlelyd. \* Byttet må forsvinne, og så komme igjen en liten stund senere.

### ✓ Sjekkliste

- ☐ Vi starter med å la byttet forsvinne hvis den berører JafseFisk, og så komme tilbake etter 3 sekunder. Bruk berører blokken for å sjekke om byttet kommer borti JafseFisk.



### 🚩 Test prosjektet

Prøv spillet ditt igjen. Ser du noen problemer? ? Legg merke til at byttet forsvinner uansett hvor det berører JafseFisk. Dessuten kan fisken bare vente 3 sekunder og så spise byttet i samme øyeblikk som det dukker opp igjen, det er ikke særlig rettferdig!

- Hvordan kan vi sikre at byttet bare forsvinner hvis det berører JafseFisks munn? Tja, vi kunne bruke berører farge blokken og se om den berører fiskens tenner. For å

gjøre dette, bytt ut **berører** blokken med en **berører farge** blokk i skriptet ditt, klikk på fargen i blokken og klikk så på fiskens tenner.

- Nå kan vi la byttet flytte seg til et tilfeldig punkt på skjermen før den dukker opp igjen ved å bruke en **gå til** blokk, og gi den en tilfeldig verdig for x og y.



## Test prosjektet

Prøv spillet igjen. Forsvinner byttet bare når det berører fiskens tenner? Og kommer det igjen i et tilfeldig punkt på skjermen? altså ikke samme sted som det ble spist?

- JafseFisk må vite når den har spist noe slik at den kan gi fra seg en lyd og bytte drakt. For å gjøre dette kan vi la byttet **send melding** om at det er spist, før det forsvinner.



Nå vil vi at fiskens respons på denne meldingen er å lage en gomme lyd og klikke med kjevene.

- Legg til drakten Dyr/shark-a og lyden Effekter/bubbles til JafseFisk. Kall drakten lukket munn.
- Legg så til et nytt skript til JafseFisk slik at han kan svare på meldingen **send melding** fra byttedyret. Dette skriptet gjør at fisken spiller av boblelyden og **byter drakt til** of lukket-munn drakten, venter litt og så bytter tilbake.



Nå er JafseFisk klar til å spise, så la oss fylle havet med byttedyr. Høyreklikk på byttefiguren og velg **lag kopi** til du føler du har fått nok fisk.

---

## Test prosjektet

Klikk på det grønne flagget. Spiser JafseFisk byttet? Spiser den alle byttedyrene?

---

## Lagre prosjektet

### Noe å tenke på

Hvorfor må vi legge til en **vis** blokk på starten av byttedyrets skript? Tenk på hva som ville skje om byttet blir spist og spillet stoppes før det dukker opp igjen. Og hva ville skje om spillet ble startet igjen?

Godt gjort! Du har igrunn fullført spillet! Men fins flere muligheter for utvidelse av spillet. Er du klar for en utfordring?

## Utfordring 1: Forandre bevegelsene til byttedyrene

For øyeblikket beveger alle byttedyrene seg likt. Kan du få ett av dem til å bevegeseg annerledes? Hint: Ikke bruke for lang tid på denne oppgaven uten å se på de andre aktivitetene i dette prosjektet.

Velg deg ut et byttedyr å eksperimentere med. Hvis de har samme drakt, bytt farge med **sett farge effekt** blokken. Slik kan du se forskjell fra resten av byttedyrene. Prøv nå å få dette byttedyret til å bevege seg saktere enn de andre.

Få byttedyret til å bevege seg saktere enn de andre. Hint: Se på blokken **gå (2) steg**.



### Test prosjektet

Beveger byttet seg saktere? Gjør dette spillet bedre? Hvis du klarte dette, , prøv å gjøre et av byttedyrene kjappere enn de andre.

Beveger byttedyrene seg på en fornuftig måte? Gjør disse forandringene spillet bedre? Hint: Hvis byttet ditt svømmer rundt i sirkler, sjekk verdiene i **tilfeldig tall** blokken i **vend** blokken.

Hva om du lar alle byttedyrene bevege seg forskjellig, ved å bruke forskjellige kombinasjoner av disse bevegelsene?

Gjør noen av disse forandringene spillet bedre? Gjør de spillet med interessant, morsommere, vanskeligere eller lettere? Er noe av dette bedre, syns du?



### Lagre prosjektet

## Utfordring 2: Hjelp byttet å unngå JafseFisk

Byttedyrene i dette spillet er skikkelig dumme! De svømmer bare tilfeldig rundt til de blir spist. Ekte fisk svømmer vekk fra rovfisker. Nå vil vi la ett av byttedyrene svømme vekk fra JafseFisk.

Det fins ingen blokk i Scratch som kan gi oss retningen til en annen figur. Men du kan få en figur til å snu seg i retningen av en annen, og deretter la den snu seg i motsatt retning. Blokkene du trenger er i **Bevegelse** paletten.

Prøv nå å hjelpe et av byttedyrene med å snu seg vekk fra JafseFisk. Du vil kanskje også la den virre mens den svømmer bort? Du vil kanskje oppdage at byttet setter seg fast i et hjørne? Du ønsker kanskje at byttet bare ønsker å flykte dersom JafseFisk kommer for nære? Hint: Se tilbake på hvordan vi brukte **avstand til** blokken tidligere i spillet.



## Test prosjektet

Gjør dette at fisken er vanskeligere å ta? Gjør det spillet bedre?



### Lagre prosjektet

#### Utfordring 3: Legg til poeng

Det er ikke nok bare å spise fisk. Hvordan vet du at du er en bedre spiller enn vennene dine? Du må kunne samle poeng, så la oss legge til en poengtavle. Se på Keep Score scratch card for å få noen hint om hvordan det kan gjøres. Pass på at poengene går tilbake til null ved begynnelsen av spillet. Hvor skal du legge inn denne blokken?



## Test prosjektet

Går poengsummen tilbake til null når spillet starter? Går den opp hver gang du spiser byttedyr?



### Lagre prosjektet

#### Utfordring 4: Legg til en nedtelling

Gi deg selv en tidsfrist. Hvor mange fisk kan du spise på 30 sekunder?

Se på Timer scratch card for å se hvordan man legger til en tidtaker til et spill. Begynn med 30 sekunders-spill.



## Test prosjektet

Begynner tidtakeren på 30? Teller den ned med rett hastighet? Kan du fange fisk mens tiden telles ned? Stopper spillet når telleren når null?



### Lagre prosjektet

## Utfordring 5: Legg til bonuspoeng

Legg til en belønning med høy bonus poeng om du kan spise alle tre fiskene samtidig. Hvordan kan du vite hvor mange som er spist?

Hint: En måte å gjøre dette på er å bruke en variabel for å telle hvor mange byttedyr som svømmer i havet.



Lagre prosjektet

## Utfordring 6: Forandre spillet: Hold byttedyrene i live!

Av og til kan man få glimrende nye idèer ved å gjøre det motsatte av det man allerede har gjort.

Endre spillet slik at du isteden kontrollerer et byttedyr i et hav av mange JafseFisk. Hvor lenge kan du holde det gående før du blir spist? Istedet for å bruke poeng, hva med å gi byttedyret 3 liv og avslutte spillet når de er brukt opp?



Lagre prosjektet

Godt gjort, du er ferdig! Nå kan du nyte spillet ditt! Ikke glem at du kan dele spillet med alle vennene og familien din ved å klikke på Legg ut i topp-menyen!

Dette kurset er utviklet av [Code Club \(http://www.codeclub.org.uk/\)](http://www.codeclub.org.uk/) i England, og oversatt til norsk av [Kodeklubben \(http://www.kodeklubben.no/\)](http://www.kodeklubben.no/).

Dette kurset er utviklet av [Code Club \(http://www.codeclub.org.uk/\)](http://www.codeclub.org.uk/) i England, og oversatt til norsk av [Kodeklubben \(http://www.kodeklubben.no/\)](http://www.kodeklubben.no/).

## Ørkenløp

### Introduksjon

Dette er et spill for to, der en papegøye og en løve kjemper om å komme først gjennom ørkenen. Hver spiller må trykke en tast så fort og ofte som mulig for å flytte figuren sin, og den som kommer først til kanten av skjermen vinner.





skjermbilde

## Steg 1: Lage en scene og legg til figurer



## Sjekkliste

- ☐ Klikk på Scene og hent en ny bakgrunn fra biblioteket. Velg Natur/desert.
- ☐ Fjern Sprite1 ved å høyreklikke på figuren og velg Slett.
- ☐ Legg til en ny figur ved å trykke på . Velg Dyr/Lionness.
- ☐ Legg så til enda en ny figur: velg Dyr/Parrot. Krymp figuren slik at den er omtrentt like stor som løvinnen ved å bruke .

## Steg 2: La løven og papegøyen bevege seg

Vi vil at figurene skal bevege seg når du trykker på en knapp.

## Sjekkliste

- ☐ Velg først løvefiguren og få den til å **gå (4) steg** når du trykker L tasten. **blocks** når [l v trykkes] **gå (4) steg**
- ☐ Velg så papegøyefiguren og la den **gå (4) steg** når du trykker A tasten.



---

## Test prosjektet

Trykk på det grønne flagget. Beveger løven og papegøyen seg over skjermen når du trykker på A og L tastene?

---

## Lagre prosjektet

## Steg 3: Start kappløpet

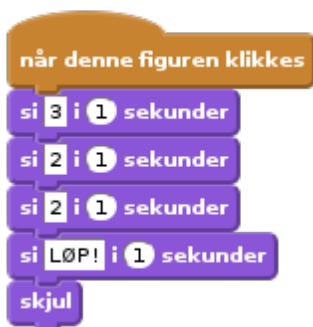
Nå må vi kjøre i gang kappløpet og kåre en vinner. Vi begynner med å lage startknapp.

## Sjekkliste

- ☐ Legg til en ny figur. Velg Ting/Button3. Flytt den til midten av scenen.
- ☐ Klikk på Drakter-fanen og symbolet T for å legge til tekst. Trykk på venstre kant av knappen for å legge til et tekstfelt og skriv inn teksten start. Du kan flytte på teksten ved å trykke en gang på den, og endre innhold ved å dobbeltklikke.
- ☐ Legg nå til et skript som viser figuren når spillet starter:



- ☐ I tillegg vil vi ha en knapp som teller ned fra 3, sier LØP og deretter blir skjult når den klikkes. Dette ordner du med følgende skript:



---

## Test prosjektet

Klikk på det grønne flagget og trykk på startknappen. Teller knappen ned? Sier den LØP? Blir den borte?

---

## Lagre prosjektet

Vi ønsker at figurene bare beveger seg etter at kappløpet er startet og vi ønsker å vite når kappløpet er over.

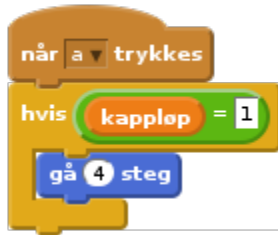
- For å vite når kappløpet har startet og sluttet lager vi en variabel med navnet **kappløp**. Variabelen skal være tilgjengelig for alle figurer. Fjern avhukingen foran variabelen, slik at den ikke vises på scenen.
- Sett kappløp til [0] når spillet startes. Forandre **når flagget klikkes** skriptet slik:



- Når nedtellingen er ferdig og løpet begynner, forandrer vi kappløpverdien til 1.

Dette gjør du ved å legge til blokken **Sett [kappløp] til (1)** under **si [1] i (1) sekunder** i skriptet som starter med **når denne figuren klikkes**.

- Så må vi lage en regel som sier at figurene bare får lov til å bevege seg når løpet har startet det vil si når kappløp har verdien 1. Klikk først på papegøyen. Så legger du til:



- Gjenta det samme for løvinnen.



## Test prosjektet

Klikk på det grønne flagget.

Kan løven og papegøyen bare flytte seg når nedtellingen er ferdig?



## Lagre prosjektet

# Steg 4: Avslutte kappløpet

Nå vil vi vite hvem som vinner kappløpet, og i tillegg gjøre klart for en ny runde.



## Sjekkliste



Legg til en blokk i papegøyens skript som **setter [kappløp] til (0)** hvis figuren berører kanten av skjermen.



Spill så inn en lyd som skal avspilles hvis papegøyen vinner. Trykk på Lyder-fanen og deretter mikrofon-ikonet og spill inn en morsom trudelutt! Opptaket starter når du har klik

på rundingen slik at den blir rød. Klikk Stopp (firkanten) når du er ferdig, og gi den et navn for eksempel Polly. Noen nettlesere kan spørre om tillatelse til å spille inn lyd. Hvis du ikke ønsker dette, bruk lydene som følger med figurene.



Deretter legger du til blokkene som vil spille lyden og la papegøyen fortelle at den vant:



Gjør det samme for løvinnen.



## Test prosjektet

Klikk på det grønne flagget.

Kan du trykke på startknappen og deretter bevege dyrene med tastene A og L? Kommer riktig vinnerlyd og melding opp på skjermen?



Lagre prosjektet

## Steg 5: Nullstill spillet

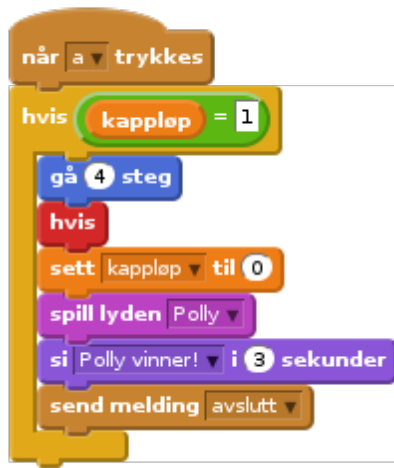
Når kappløpet er over må vi fortelle de andre figurene at spillet er over og nullstille spillet, slik at er klart for en ny runde.



## Sjekkliste



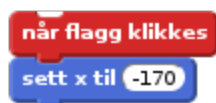
Klikk på papegøyefiguren og legg til en blokk som sender melding avslutt etter at figuren sier den har vunnet.



- ☐ Vi trenger nå et nytt skript som lytter etter denne avslutningsmeldingen og flytter papegøy tilbake til start.



- ☐ Legg til det samme skriptet for løven. Test forskjellige x-verdier for å være sikker på at løven og papegøyen starter fra samme sted.
- ☐ For at figurene skal stå på startstreken når kappløpet starter den aller første gangen må vi også legge til følgende blokk til begge figurene:



- ☐ For at spillerne skal kunne klikke i gang nye runder må passe på at start-knappen kommer etter hver avsluttet runde. Klikk på figuren og legg til et skript som viser knappen når avslutningsmeldingen er mottatt.



---

## Test prosjektet

Klikk på det grønne flagget.

Kan du spille mot en venn? En av dere styrer papegøyen ved å trykke A, og den andre løven ved å trykke L.

## Utfordring 1: Legg til en rakett!

- ☐ Legg til en rakett som kan brukes én gang per kappløp og som flytter papegøyen eller løven 30 steg på en gang.
- ☐ Legg til en ny drakt med ild som kommer ut bak hver figur. La denne aktiveres når raketten avfyres.
- ☐ Lag en lyd som figuren kan gi fra seg når raketten avfyres.

``blocks når [a v] trykkes hvis <(kappløp) = (1) og (rakett) = (0)> skift til drakt [parrot-b v] sett [rakett v] til (1) gå (4) steg spill lyden [Rooster v] hvis sett [kappløp v] til (0) spill lyden [Polly v] si [Polly vinner! v] i (3) sekunder send melding [avslutt v]



Test prosjektet



Lagre prosjektet

## Utfordring 2: Bruk egendefinerte blokker for å forenkle skriptet ditt

Koden som brukes til å sjekke om kappløpet er over brukes nå to steder for hver figur; når figuren beveger seg normalt og når den beveger seg med rakett. Vi kan forenkle skriptet vårt ved å bruke en egendefinert blokk. Dette er en samling kode som brukes flere steder. Det er nesten som at vi lager vår egen Scratch kodeblokk!

- ☐ Velg papegøyens skript.
- ☐ Velg Flere klosser-paletten og klikk så på Lag en kloss knappen.
- ☐ Kall klossen din ferdig og trykk OK.
- ☐ Du vil nå få en **definer ferdig** blokk i skriptvinduet ditt. Flytt den litt for seg selv.
- ☐ Løsriv **hvis berører [kant v]?** blokken og dra den til **definer ferdig** blokken.



Kan du dra **ferdig** blokken fra paletten og bruke den slik som andre kodebiter?

Slett den andre **hvis berører [kant v]?** blokken fra skriptet ditt og erstatt den med en **ferdig** blokk.

Gjør dette koden din enklere å lese? Kan du lage en tilsvarende egendefinert blokk for løvinnen?



Test prosjektet



## Lagre prosjektet

Veldig bra! Nå er du ferdig og kan nye spillet du har laget!

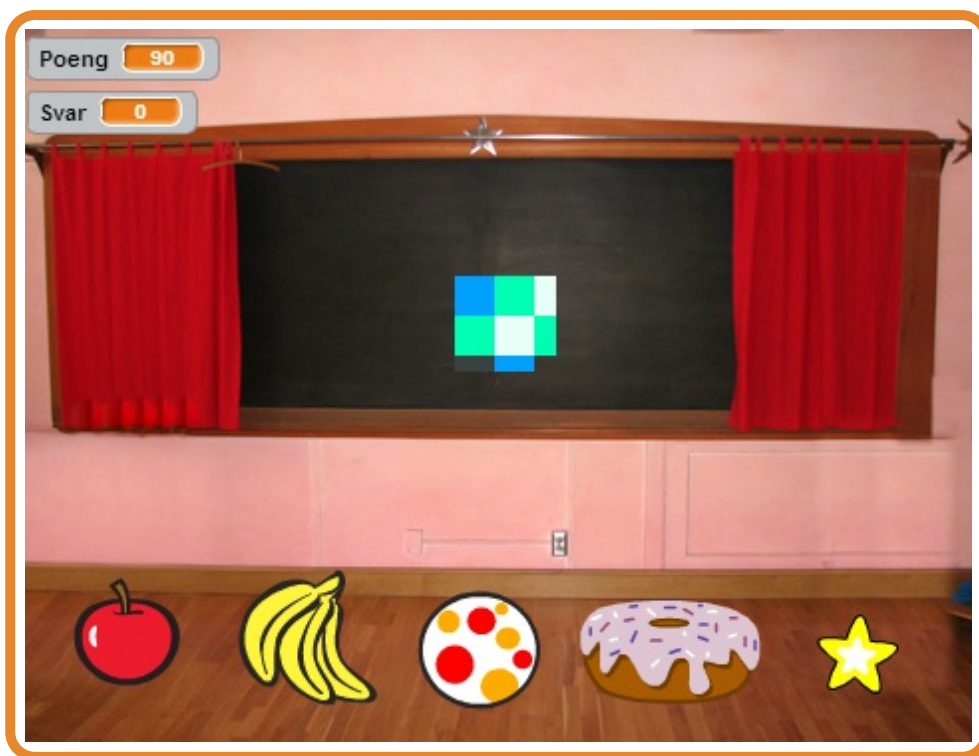
Dette kurset er utviklet av [Code Club \(http://www.codeclub.org.uk/\)](http://www.codeclub.org.uk/) i England, og oversatt til norsk av [Kodeklubben \(http://www.kodeklubben.no/\)](http://www.kodeklubben.no/).



Dette kurset er utviklet av [Code Club \(http://www.codeclub.org.uk/\)](http://www.codeclub.org.uk/) i England, og oversatt til norsk av [Kodeklubben \(http://www.kodeklubben.no/\)](http://www.kodeklubben.no/).

## Introduksjon

Et bilde av en tilfeldig ting vises på tavlen. Men bildet er forvrengt, slik at du må gjette hva det er ved å klikke på et av alternativene som vises under. Desto raskere du gjetter riktig, desto flere poeng får du.



skjerm bilde

## Steg 1: Få flere ting til å vise seg på tavlen


Vi vil at noen forskjellige bilder skal komme opp på tavlen.



Sjekkliste



Start et nytt Scratch-prosjekt og slett kattefiguren.

Klikk på Scene og deretter Bakgrunner-fanen. Åpne biblioteket med bakgrunner ved å trykke på  og velg så Innendørs/chalkboard.

Importer en valgfri figur. Velg gjerne en fra Ting-mappen.

Plasser figuren på midten av tavlen, og endre størrelsen hvis den ikke passer.

Legg til fire nye drakter fra Ting-mappen. Du kan velge hva du vil!

La oss nå få en tilfeldig ting til å dukke opp. Bruk dette skriptet.



## Test prosjektet

Trykk på det grønne flagget. Endrer figuren seg? Klikk flere ganger. Får figuren stadig nye drakter? Flott.

Det gjør ingenting om samme drakt kommer opp flere ganger på rad. Det er helt normalt når det trekkes en tilfeldig drakt hver gang. Du legger kanskje merke til at det flimrer litt når drakten skiftes? Det skal vi fikse i neste steg.



## Lagre prosjektet

# Steg 2: Forvrenge bildet

La oss nå forvrenge figuren når den dukker opp på tavlen, så det blir vanskeligere å gjette hva det er. Deretter skal vi gradvis gjøre vi den tydeligere igjen.

Vi skal bruke en poeng-variabel til å kontrollere graden av forvrenging. Dersom poengscoren er høy vil bildet bli veldig forvrengt. Når antallet poeng synker, vil også graden av forvrenging synke. Poengvariabelen fungerer dermed som en tidteller.



## Sjekkliste



Velg Data-paletten og opprett en variabel kalt poeng.

Endre skriptet slik:



---

## Test prosjektet

Trykk på det grønne flagget.

Kommer det opp et tilfeldig og forvrengt bilde?

Blir bildet gradvis tydeligere?

Går poengsummen ned i takt med at bildet blir tydeligere?

Blir bildet fullstendig tydelig når poengsummen er 0?

Får du fremdeles nye ting på tavlen når du klikker på det grønne flagget?

---

## Lagre prosjektet

### Ting å prøve

- ☐ Prøv å endre poengsummen fra start, samt hvor mye den skal forandre seg for hver gang den går gjennom løkka. Hvordan endrer deg utseendet til bildet? Blir det vanskeligere eller enklere å se hva bildet forestiller?
- ☐ Forsøk noen ulike grafiske effekter fra nedtrekkslisten. Hvordan påvirker dette vanskelighetsgraden?

## Steg 3: La spilleren prøve å gjette bildet

Så langt har vi fått vårt tilfeldige bilde til gradvis å bli tydeligere, samtidig som poengsummen synker. Men hvordan skal man vinne spillet? Vi vil legge til noen figurer nederst på skjermen som spilleren kan klikke på. Klikker man på den rette, vinner man spillet. Klikker man på feil, forsvinner figuren og spillet fortsetter med en ny figur.

Først må vi å vite hva det rette svaret er.

### Sjekkliste

- ☐ Opprett en ny variabel og kall den svar. Pass på at den er tilgjengelig for alle figurer.
- ☐ Endre skriptet slik at det klarer å holde styr på hva som er rett svar. Etter den første **gjenta** løkken legger du derfor til blokken **sett [svar] til drakt #**:

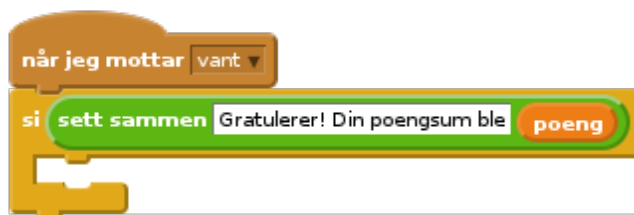


- ☐ Nå må vi legge til flere figurer som spilleren kan klikke på.
- ☐ Gi først figuren din navnet spørsmål.
- ☐ Lag så en kopi av figuren ved å høyreklikke på den. På scenen drar du deretter figuren ned til det venstre hjørnet.
- ☐ Endre figurens navn til svar1.
- ☐ Slett skriptet til svar1 og alle kostymer bortsett fra det første.
- ☐ Gjenta de tre siste stegene igjen (kall neste kopi svar2), men plasser svar2 ved siden av svar1 og slett alle bortsett fra den andre drakten.
- ☐ Gjenta disse punktene tre ganger til, slik at du har fått laget svar3, svar4 og svar5. Du skal ha en rad med fem svar-figurer nederst på scenen, hver viser en drakt som hovedfiguren kan ha. Ingen av svar-figurene skal ha skript knyttet til seg.
- ☐ Nå må vi få alle figurene til å reagere når de blir klikket på. Hva som skal skje avhenger av

om spilleren har klikket riktig eller galt. Legg til dette skriptet til svar1:



- ☐ Dra skriptet over til de andre figurene, slik at alle får hver sin kopi. For hver figur, bytt 1 til 2, 3, osv.
- ☐ Nå skal vi lage skriptet som gir melding til spilleren når han har vunnet. Klikk på spørsmål igjen og legg til dette skriptet:



---

## Test prosjektet

Trykk på det grønne flagget.

Når du tester spillet kan du bruke svarskjermen på scenen for å si hva rett svar er. Det fungerer bra for testing.

Hva skjer når du klikker på riktig svar?

Hva skjer når du klikker på galt svar?

Hva skjer med det gale svaret når du starter på et nytt spill?

Testen viser oss to problemer: Først og fremst, ting som ble klikket på ved galt svar kommer ikke tilbake når et nytt spill starter. For det andre, poengsummen fortsetter å gå ned, selv når man har klikket på riktig svar.

- For å fikse det første problemet må vi legge til følgende skript for hver av de fem svarfigurene:



For å fikse det andre problemet må vi få stoppet spørsmålfigurens **gjenta til**-løkke, når spilleren

klikker på riktig svar. Vi kan bruke en ny variabel for å gjøre det. Vi kaller denne vant og legger inn en **sett**-blokk som gir den verdien 0 når spillet starter, og en annen som setter verdien til 1 når spillet vinnes. Se skript under. + I tillegg må vi stoppe **gjenta til**-løkken når poengsummen har blitt 0 eller vant er 1. + Til slutt legger vi også inn en **ta bort grafiske effekter**-blokk som avslører spørsmålsfiguren, når spilleren har gjettet riktig. Skriptet skal nå se slik ut:

```
```blocks
når grønt flagg klikkes
  skjul
  gjenta <tilfeldig tall fra(1) til (5)> ganger
    neste drakt
  slutt
  sett [svar v] til (drakt #)
  sett [poeng v] til (110)
  sett [vant v] tuk (0)
  gjenta til <<((poeng) = (0)> eller <(vant) = (1)>>
    endre [poeng v] med [-10]
    sett [piksel v] effekt til (poeng)
    sett [farge v] effekt til (poeng)
  vis
  vent (1) sekunder
  slutt

  når jeg mottar [vant v]
    sett [vant v] til (1)
    ta bort grafiske effekter
    si (sett sammen [Gratulerer! Din poengsum ble] (poeng))
```
```



## Lagre prosjektet

Gratulerer! Du er nå ferdig med spillet. Men det fins mange flere ting du kan gjøre med det. Prøv deg gjerne på utfordringene vi har laget!

## Utfordring 1: Gjør spillet enklere eller vanskeligere

Endre vanskelighetsgrad for spillet.



- Forsøk å endre hvor lenge bildet vises frem og hvor raskt poengsummen minker.
- Forsøk å endre forvrengingen av bildet.
- Forsøk å gjøre tingene likere hverandre eller mer forskjellig. Husk også å forandre svarfigurenes drakter.



Lagre prosjektet

## Utfordring 2: Forvreg bildet ulikt fra gang til gang

For øyeblikket bruker spillet samme forvrengingsalgoritme hele tiden. Men i steg 2 prøvde du kanskje ut noen forskjellige alternativer. Prøv nå om du kan finne noen flere forvrenginger som du synes virker like bra som farge og piksler.

Endre spillet slik at hvert spill bruker forskjellige forvrengninger i gjenta til-løkken.

Hint: Forsøk å opprette en ny variabel som du kaller forvrenging. Sett denne til en tilfeldig verdi i starten av spillet. Bruk så hvis-blokker i gjenta til-løkken for å velge ut en forvrenging til det hvert spill.



Lagre prosjektet

## Utfordring 3: La hvert spill ha flere runder

For øyeblikket er hvert spill uavhengig av andre. Prøv om du kan legge til flere runder slik at man får gjette på tre ting og kan vinne inntil 300 poeng.

Hint: Du vil trenge en ekstra variabel for å lagre den totale poengsummen. Du må også ha en løkke som går rundt for hver runde.

Hint: Du vil trenge en ekstra variabel for å lagre den totale poengsummen. Du må også ha en løkke som går rundt for hver runde.



Lagre prosjektet

## Utfordring 4: Øk vanskelighetsgraden gradvis

Gjør nå spillet vanskeligere og vanskeligere for hver runde.

Kanskje hver runde også skal gi ulikt antall poeng? Bør spilleren også få ekstra mange poeng for å gjette kjapt i de vanskeligste rundene?

Hint: : Hvordan kan du vite hvilken runde du er i? Hvordan kan du bruke det til å endre vanskelighetsgraden og poengsummen?



Lagre prosjektet

## Utfordring 5: Fortsett til spilleren gjør feil

I stedet for et bestemt antall runder, kan du la spillet gå til det blir klikket på feil svar. Dette fungerer nok best dersom man også øker vanskelighetsgraden utover i spillet.



Lagre prosjektet

## Utfordring 6: Gjør spillet enklere eller vanskeligere basert på hvor flink spilleren er

I stedet for å gjøre det stadig vanskeligere kan vi tilpasse vanskelighetsgraden til spillernes dyktighet. Hvis de raskt gjetter riktig ting, kan den neste runden gjøres vanskeligere. Hvis de klikker feil eller gjetter sakte, kan neste runde gjøres enklere.

Dette fungerer bare hvis du ikke samler opp poengsummen fra runde til runde.



Lagre prosjektet



## Utfordring 7: Hold styr på rekorden

Finn en måte å lagre den høyeste poengsummen på. Klarer du også å lagre navnet til spilleren, og få spillet til å si hvem som har rekorden?



Lagre prosjektet

## Utfordring 8: Gi en straff for galt svar

Slik spillet er nå kan man bare klikke som en gal på alle svarene, så vil man raskt finne riktig svar. Det kan derfor være en god idé å trekke fra poeng hver gang spilleren klikker galt.

Gjør dette spillet bedre?



Lagre prosjektet

Veldig bra! Nå er du ferdig og kan nye spillet du har laget! Ikke glem å del spillet ditt med venner og familie ved å trykke på Legg ut i menyen!

Dette kurset er utviklet av [Code Club \(http://www.codeclub.org.uk/\)](http://www.codeclub.org.uk/) i England, og oversatt til norsk av [Kodeklubben \(http://www.kodeklubben.no/\)](http://www.kodeklubben.no/).



Dette kurset er utviklet av [Code Club \(http://www.codeclub.org.uk/\)](http://www.codeclub.org.uk/) i England, og oversatt til norsk av [Kodeklubben \(http://www.kodeklubben.no/\)](http://www.kodeklubben.no/).

## Introduksjon

Dette prosjektet lager tegneprogram slik at du kan lage din egen kunst. Du kan tegne med forskjellige farger, bruke viskelær, lage stempler og mye mer!



skjerm bilde

## FORBEREDELSE: Last ned nødvendige ressurser.

Denne første delen kan du godt få hjelp fra en voksen til å gjøre!



### Sjekkliste

- ☐ Lag en ny tab i nettleseren din
- ☐ Skriv inn denne nettadressen: <http://bit.ly/Tegneutstyr>
- ☐ Last ned zip-fila "Ressurser\_Tegneprogram.zip" og legg det på skrivebordet på datamaskinen din eller en annen plass du husker.
- ☐ Pakk ut zip-filen ved å høyreklikke på den og velge "Extract All"/"Pakk ut filer".

## Steg 1: Dra og tegn!

Vi starter med en penn som tegner når du drar den rundt på scenen.

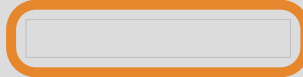
Start et nytt Scratch-prosjekt. Slett katten ved å høyreklikke og velge slett.

Klikk på Scene og så på Bakgrunner-fliken. Last opp bakgrunnen som heter frame.bmp fra folderen du hentet i forberedelsene.

Lag en figur som heter penn av bildet green-pencil som ligger i den nedlastede mappen. Bytt til drakter-fliken og velg senterpunkt for figurer ved hjelp av



. Flytt blyanten slik at spissen peker på det lille grå korset i midten. Da vil spissen tegne og ikke midten av blyanten.



Få pennen til å følge musa rundt på scenen ved å bruke for alltid-blokka og gå til musepeker-blokka.



Nå vil vi bruke denne pennefiguren som en ordentlig penn. Om du ser under penn-kategorien kan du se alle slags tegefunksjoner. De vi er interessert i nå er penn på og penn av

Vi vil bruke museknappen til å kontrollere pennen - når museknappen er nede går pennen ned og når museknappen er oppe er pennen oppe. Vi kan gjøre dette ved å bruke enn hvis ... ellers ... og en mus nede-blokk



## Test prosjektet ditt

Klikk på det grønne flagget. Følger pennen musa rundt? Hva skjer om du holder museknappen

nede og flytter på musa? Ikke bry deg om pennefargen enda.

### Sjekkliste

- ☐ Etterhvert vil skjermen bli ganske full av rabbel. Vi kan bruke slett-blokka til å fjerne dette.



## Test prosjektet ditt

Klikk på det grønne flagget.

Forsvinner tegningene dine når du klikker på det grønne flagget?

---

### Lagre prosjektet

## Steg 2: Rydde opp

I stedet for å måtte starte og stoppe prosjektet for å slette tavla kan vi lage en knapp som sletter alt i stedet. Vi kan bruke slett-blokka.

### Sjekkliste

- ☐ Lag en ny figur fra ressursmappa du lasta ned i begynnelsen av prosjektet. Velg cancel-button.
- ☐ Bytt navn på figuren til slett.
- ☐ Flytt figuren til nederste høyre hjørne av scenen.
- ☐ Gi slette-figuren dette skriptet:



## Test prosjektet ditt

Klikk på det grønne flagget.

Sletter sletteknappen alle tegningene dine?



Lagre prosjektet

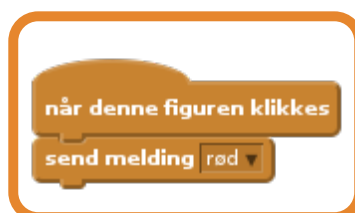
## Steg 3: Bytte farge

Til nå har vi bare kunnet tegne blå streker. Vi kan bruke andre farger også! Vi legger til noen figurer i bunn av rammen. Figurene vil se ut som fargede knapper. Når vi klikker på en knapp endres pennefagen til den fargen knappen har. For å vite at vi har byttet farge skal vi gjøre det slik at blyanten skifter farge til den fargen vi bruker.



### Sjekkliste

- ☐ Hent en ny figur fra ressurser/red-selector.
- ☐ Gi den navnet Rød og flytt den ned i venstre hjørne
- ☐ Gi den et skript som sender meldingen rød.



Dette er alt den gjør. Det vanskelige arbeidet gjøres av blyanten.

- ☐ Klikk på blyanten og importer drakten Ressurser/red-pencil . Sett midtpunktet til blyantspissen for denne drakten også.
- ☐ Legg til et nytt skript. Når blyanten får meldingen rød, skal den også forandre drakt til rød (red-pencil). Og så skal den selvsagt også begynne å tegne rødt. Slik bygger du skriptet:

Hint: Når du skal velge farge kan du flytte dråpeplukkeren bort til fargeblyanten og velge rødfargen derifra også.



## Test prosjektet ditt

Klikk på det grønne flagget.

Begynn å tegne en strek. Bytt til rødt, og se om fargen forandres. Kommer streken fra blyantspissen nå også?



Lagre prosjektet

Gjenta punktene for å lage blå, gule og grønne knapper.

## Test prosjektet ditt

Klikk på det grønne flagget.

Funker alle knappene? Skifter de til en annen farge på blyanten? Tegner de med riktig farge? Tegner alle figurene fra tuppen av blyanten?



Lagre prosjektet

## Steg 4: Bare tegne på tavla

Du har sikkert lagt merke til at man kan tegne over hele scenen, og det blir jo litt rotete. Skal vi

begrense skriblingen til bare den lysegrå tavla må vi sette grenser for hvor blyanten kan gå.

Du husker kanskje at Scratch definerer punkter på scenen ved hjelp av en x- og en y-akse. Når du flytter musepekeren rundt omkring vil du se disse verdiene nede i det venstre hjørnet. For å finne ut hvor grensene for tavla går kan vi begynne nede i det venstre hjørnet. Da står det x: - 230 og y: 120. Flytter vi pekeren rett bort til det høyre hjørnet ser vi at y er uforandret, mens x har blitt til 230. Da vet vi at x-aksen er -230 til 230. y-aksen finner vi ved å flytte pekeren opp i toppen av tavla. Da finner vi ut at y-aksen er 170 til -120.

Disse verdiene kan vi bruke in en **hvis**-blokk, og si at når musepekeren er utenfor tavlas x- og y-akse, så virker ikke blyanten

## ✓ Sjekkliste

- ☐ Fyll inn sjekkene som sier at blyanten får følge musepekeren bare hvis: y er større enn -120 og mindre enn 170 x er større enn -230 og mindre enn 230

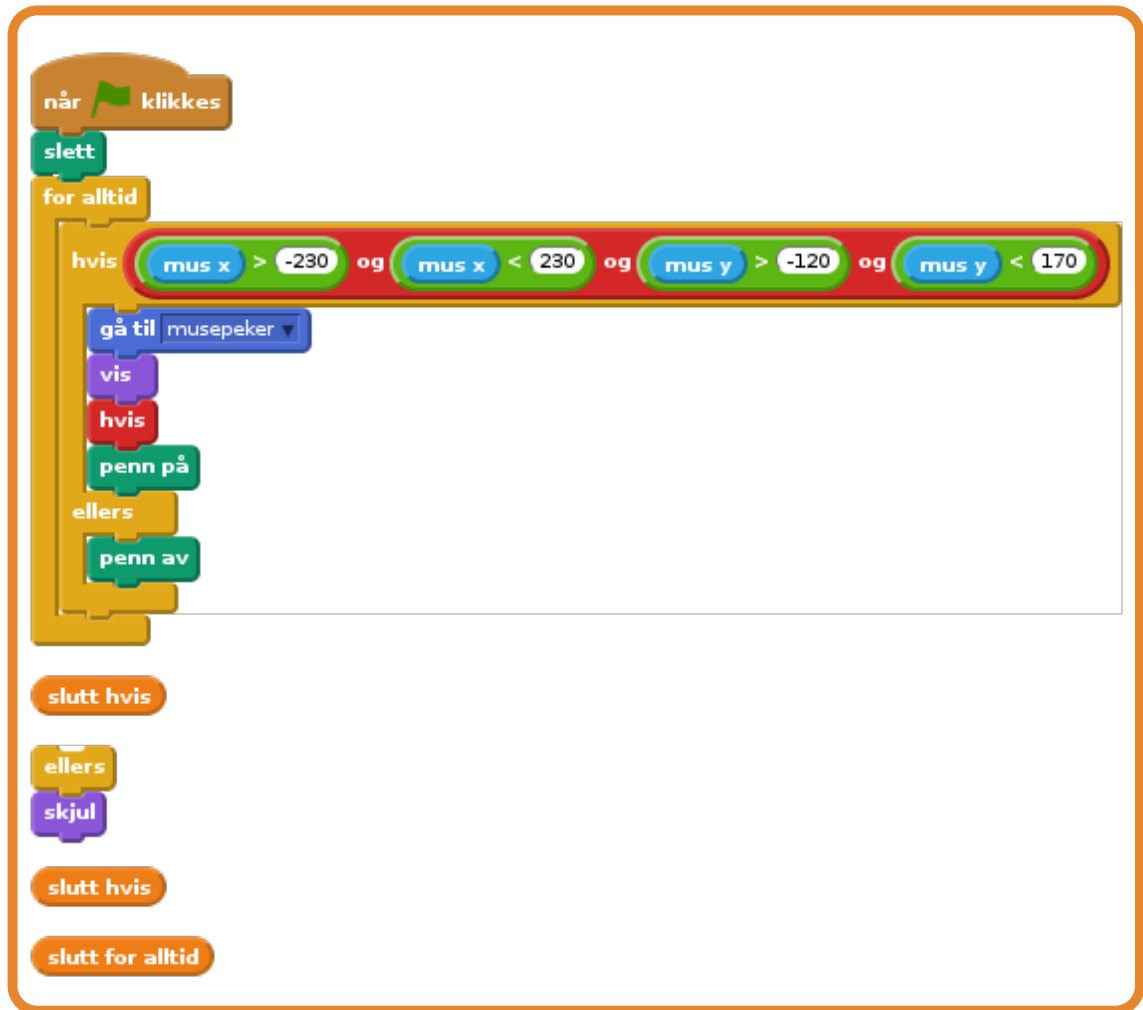
NB For å få plass til alle sjekkene må du først legge inn en **<> og <>**-blokk, og deretter legge til to nye **<> og <>**-blokker inni denne.



- ☐ Siden vi ikke kan tegne utenfor tavla er det like greit at blyanten bare blir borte når musepekeren går utenfor tavlen. For å gjøre dette må vi erstatte **hvis**-blokka ovenfor med en **hvis.. ellers..**-blokk. Reglene blir nå: hvis musepekeren er innenfor tavlas x-

og y-akse følger blyanten pekeren. ellers skjules blyanten.

NB! Fordi blyanten vil skjules når pekeren går utenfor tavla, må vi få den til å vise igjen når den går innenfor igjen. Pass derfor på at du får lagt inn en **vis**-kommando i **hvis**-blokka.



## Test prosjektet ditt

Klikk på det grønne flagget. Kan du fremdeles tegne på tavla? Kan du tegne utenfor tavla? Hva skjer med blyanten når musepekeren går ut og inn av tavla?




Lagre prosjektet

## Steg 5: Viskelær

Nå kan vi tegne hva vi vil. Men hva om vi trenger et viskelær? Hmm... da kan vi jo bare få blyanten til å tegne med samme farge som tavla! Og så gir vi blyanten en viskelærdrakt!



## ✓ Sjekkliste

- ☐ Last opp en figur fra fil. Velg figuren eraser fra mappen du lastet ned i starten. Kall figuren viskelær.
- ☐ Gjør figuren litt mindre med krympeknappen , og så drar du den ned i høyre hjørne, ved siden av slett-knappen.
- ☐ Gi viskelær-figuren et skript som sender meldingen visk.



- ☐ For å få blyanten til å viske må du legge til en viskelærer som en drakt til figuren.
- ☐ Blyanten svarer på visk meldingen med å bytte pennefarge til grå (bruk fargevelgeren for å velge bakgrunnen til tavlen). Husk å sette midtpunktet på viskelærer foran.



## Test prosjektet ditt

Klikk på det grønne flagget.

Klarer viskelæret å viske? Fungerer det helt ut til kantene av tavla? Går det greit å veksle mellom blyant og viskelær?

---

## Lagre prosjektet

### Steg 6: Stempel

Nå skal vi lage et stempel som kan lage små avtrykk på tavla.

## ✓ Sjekkliste

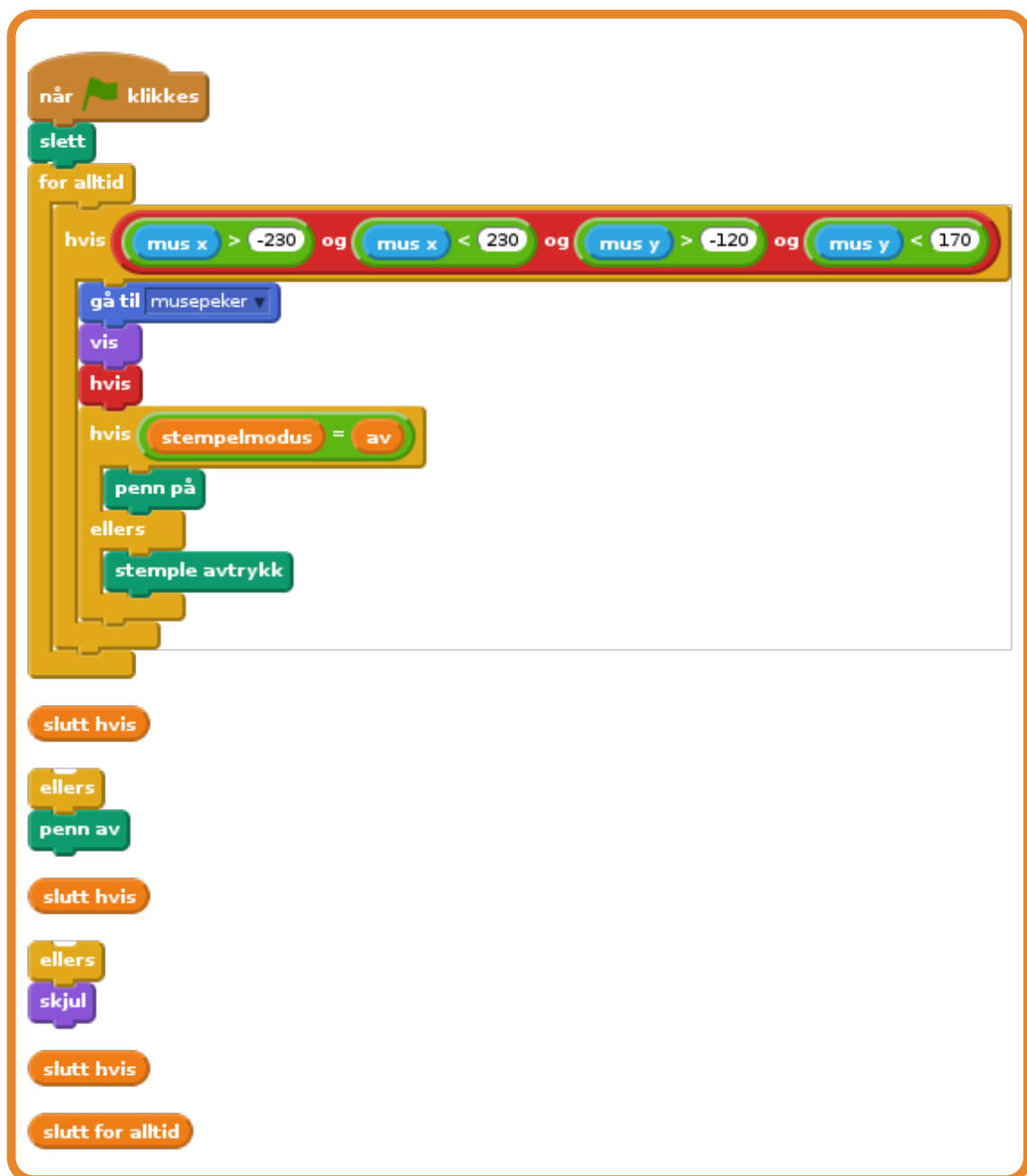
- ☐ Legg til en ny figur med valgfri drakt og kall den stempel. Vi valgte Scratch-logoen fra ting-mappen. Krymp figuren og plasser den nederst på skjermen ved siden av de andre verktøyene. Når figuren klikkes skal den sende meldingen stempel.
- ☐ Legg til en ny drakt for blyantfiguren. Det skal være samme drakt som du nettopp ga stempelet.
- ☐ Velg blyanten og legg til en ny variabel som bare gjelder for denne figuren. Kall variabelen stempelmodus. Fjern avhukingen foran variabelen slik at den ikke vises på scenen. Oppgaven til denne variabelen er å holde styr på om vi skal tegne eller stemple.
- ☐ Legg til et skript for blyanten som responderer på meldingen. Skriptet skal skifte drakt til den samme drakten du valgte for stempelet. Og så skal det sette verdien til stempelmodus til på.



- ☐ Forandre de andre skriptene som er knyttet til fargevelgerne og viskelæret slik at de setter stempelmodus til av. Eksempel (for viskelæret):



- ☐ Til slutt må vi sjekke variabelen hvis museknappen er nede? for å se om vi skal tegne eller stemple. Hvis stempelmodus er satt til på skal vi stemple, hvis ikke skal vi bruke den eksiterende penn på.



## Test prosjektet ditt

Klikk på det grønne flagget.

Klarer du å lage avtrykk? Hva skjer når du skifter tilbake til en av blyantene?



Lagre prosjektet

Veldig bra jobba! Du er nå ferdig med programmet. Prøv disse utfordringene!

## Utfordring 1: Regnbueblyant

I denne utfordringen skal du legge til en blyant som kan forandrer farge mens du tegner. Kult, ikke sant?



### Sjekkliste

Først må du legge til regnbue-valget og regnbue-kostymen:

- ☐ Legg til regnbue-selektoren som figur. Du finner den i mappen du lastet ned i starten (velg rainbow-selector.gif). Sett den ved siden av de andre selektorene nederst til venstre. Figuren skal sende meldingen regnbue når den klikkes.
- ☐ Legg til en regnbue-blyanten (\_rainbow-pencil.png) som en ny drakt til blyanten. Husk å justere midtpunktet.
- ☐ Nå må du lage et skript som får blyantfargen til å skifte mange ganger i sekundet. (Vi fant ut at å endre den med 5 hvert 0.05 sekunder fungerer fint, men du bør prøve ut egne verdier).

Hint: Scratch-kortet Timer/Tid viser hvordan du kan få noe til å forandre seg med jevne mellomrom. Men i vårt tilfelle er det pennfarge, og ikke tid, som skal forandres.

Denne blokken må legges inn i en løkke. Men du trenger også noe å kontrollere løkken med, så den bare endrer farge når regnbueblyanten er valgt.

Hint: Du kan gjøre dette på en måte som ligner på hvordan stempelmodus styrte når stampelet skulle skrus av og på. Prøv å lage en variabel som du kaller regnbuemodus. La denne skrus på når regnbueblyanten klikkes, og av når de andre velges.

## Test prosjektet ditt

Klikk på det grønne flagget.

Virker regnbueblyanten? Hva skjer når du skifter tilbake til en av de andre blyantene?



Lagre prosjektet

## Utfordring 1: Snarveier

Nå skal du få prøve å deg på å lage snarveier på tastaturet. Det betyr at man istedenfor å klikke på knappene kan bruke tastene for å bytte farge, stemple eller viske ut. Du kan bruke **hvis tast [] trykkes** for å benytte tastaturet. For hver tast du legger til trenger du en ny **hvis tast [] trykkes**-blokk som sender samme meldingene som verktøy-valgene gjør når de klikkes. Legg til skriptene til scenen.

Vi brukte disse snarveiene: \* Slett alt - a \* Visk - v \* Rød blyant - r \* Blå blyant - b \* Gul blyant - y \* Grønn blyant - g \* Regnbue blyant - w \* Stempel - s

### Test prosjektet ditt

Klikk på det grønne flagget.

Virker alle snarveiene? Virker knappene fremdeles også?



Lagre prosjektet

## Utfordring 1: Større og mindre

En annen funksjon som tegneprogrammer gjerne har er å forandre størrelsen på blyantstreken. Prøv å se om du får til dette. Det er en ting som gjør dette vanskelig. Noen ganger trenger vi å endre størrelsen på pennen og noen ganger trenger vi å forandre størrelsen på drakten. Det er avhengig av om du bruker blyanten eller stemplet.

- ☐ Lag to nye figurer ved å importere bigger-selector og smaller-selector fra mappen med ressurser. Kall figurene større og mindre.
- ☐ La figurene sende ut meldingene større og mindre.
- ☐ Blyanten kan svare på meldingen ved å enten forandre pennestørrelsen med 1 eller kostymestørrelsen med 10, avhengig av verdien på stempelmodus.

Hint: Du kan bruke en av endre-blokkene under Penn eller Utseende. For å forminske setter du bare et minustegn foran tallet.

Hint 2: For å holde styr på om det er blyantstørrelse eller stempel som skal endres må du bruke en **hvis-ellers**-blokk

- ☐ Glem ikke å lage snarveier for disse funksjonene også. For eksempel pil opp for større og pil ned for mindre.
- ☐ Har du lagt merke til at også blyantfiguren blir større når du forstørrer stempelet? For å unngå dette kan vi sette størrelsen til 100% hver gang blyant klikkes.
- ☐ For å gjøre det enda mer avansert kan du få stempelet til å huske størrelsen sin fra gang til gang. Den enkleste måten å gjøre dette på er å opprette en variabel som du kaller stempelstørrelse. Denne må oppdateres for hver gang størrelsen på stempelet endres. Når man skifter fra blyant til stempel, bestemmes så størrelsen fra denne variabelen.

## Test prosjektet ditt

Klikk på det grønne flagget.

Fungerer forstørrelsesknappen? Fungerer forminskningsknappen? Hva skjer om du bytter til stempelet, endrer størrelsen og så forandrer tilbake til blyanten?



## Lagre prosjektet

Veldig, veldig bra! Nå kan du tegne akkurat det du vil!

Don't forget you can share your game with all your friends and family by clicking on Share on the menu bar!

Dette kurset er utviklet av [Code Club \(http://www.codeclub.org.uk/\)](http://www.codeclub.org.uk/) i England, og oversatt til norsk av [Kodeklubben \(http://www.kodeklubben.no/\)](http://www.kodeklubben.no/).

## Level 2

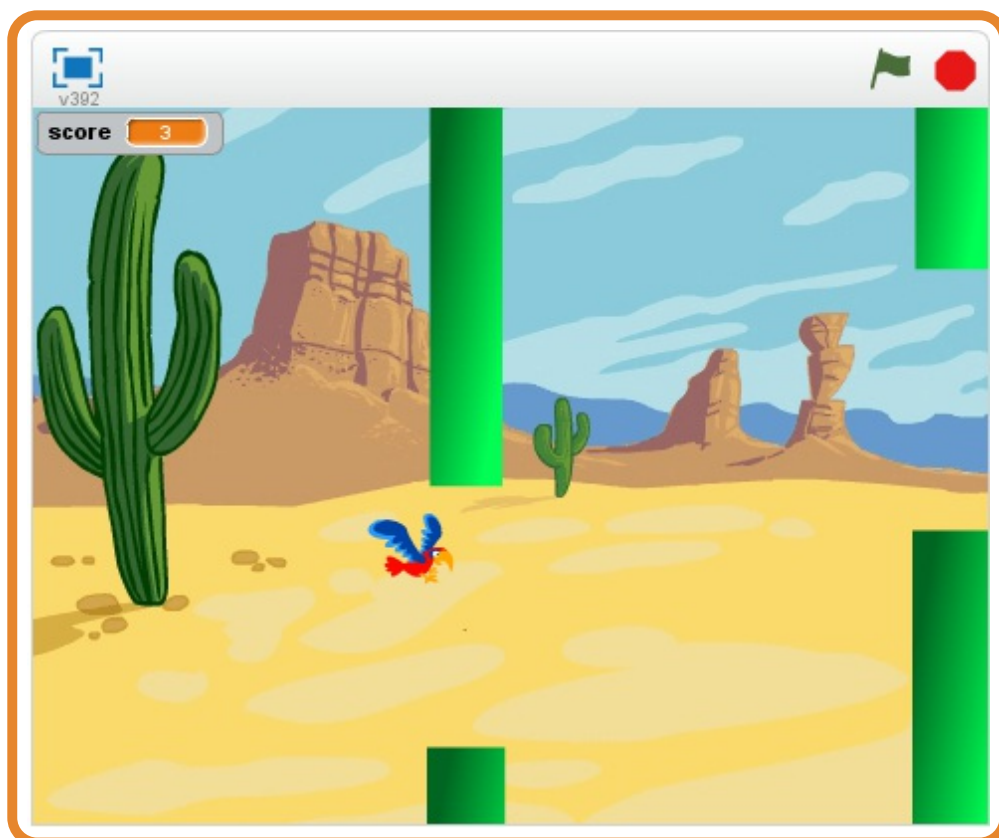
## Flaksefugl



Dette kurset er utviklet av [Code Club \(http://www.codeclub.org.uk/\)](http://www.codeclub.org.uk/) i England, og oversatt til norsk av [Kodeklubben \(http://www.kodeklubben.no/\)](http://www.kodeklubben.no/).

### Introduksjon

I denne oppgaven skal vi lage vår egen versjon av spillet Flappy Bird. Trykk på mellomromstasten for å flakse og prøv å styre mellom rørene!



screenshot

### Steg 1: Få Flakse til å falle



Sjekkliste





- Start et nytt Scratch-prosjekt. Slett katten ved å høyreklikke den og velge "Slett".
- Bytt ut bakgrunnen med et utendørslandskap. desert er et bra valg.
- Legg til Flakse-figuren. Du trenger en figur med drakter for vinger opp og vinger ned. parrot er et bra forslag.
- Bytt navn på figuren til Flakse.
- Gi Flakse dette scriptet:



## Test prosjektet ditt

klikk det grønne flagget: starter Flakse midt på skjermen og faller mot bunnen?



Lagre prosjektet ditt hvis du kan

## Steg 2: Få Flakse til å fly

Nå vil vi at Flakse skal fly oppover når du trykker mellomrom.



## Sjekkliste



- Klikk på Drakter og gi de to draktene navnene Vinger opp og Vinger ned.
- Gå tilbake til Skript og legg til dette skriptet:



## Test prosjektet ditt

Klikk det grønne flagget: klarer du å kontrollere Flakse med mellomromstasten? La du merke til at noen ganger så flytter ikke Flakse seg når du trykker mellomrom? Det er det neste vi skal fikse.

## Lagre prosjektet ditt

### Steg 3: Gjør kontrollen bedre

Vi vil at Flakse skal reagere hver gang vi trykker mellomrom. Men når vi trykker mellomrom så starter to løkker etterhverandre. Hvis vi trykker mellomrom før disse to løkkene er ferdig så skjer det ikke noe. For å løse dette problemet skal vi bruke en variabel til å telle hvor mange flaks vi trenger å gjøre.

## Sjekkliste

- ☐ Ta fra hverandre skriptet som starter med når mellomrom trykkes og legg de til siden. Vi skal bruke klossene straks.
- ☐ Lag en ny variabel `For denne figuren` og kall den `flaks`.
- ☐ Legg til dette skriptet ved å bruke blokkene du la til siden:



- ☐ Til slutt, legg til dette skriptet på når mellomrom trykkes blokken:



---

## Test prosjektet

Klikk det grønne flagget: Flakser Flakse en gang for hver gang du trykker mellomrom??

---

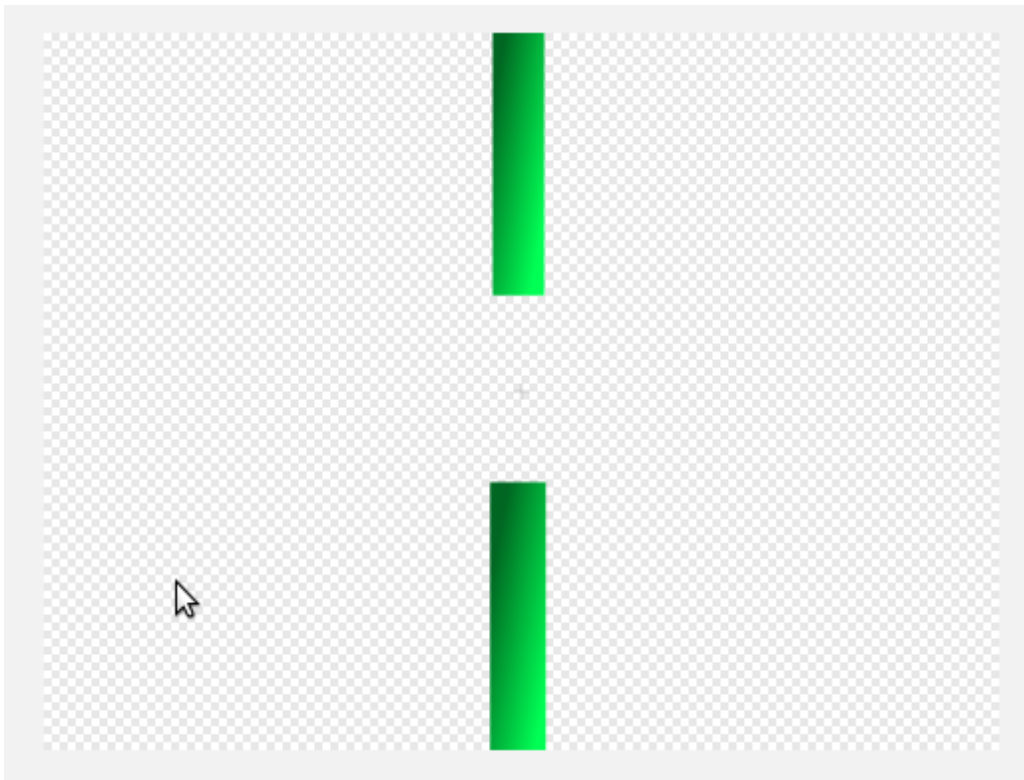
 Lagre prosjektet ditt om det trengs

## Steg 4: Legg til rørene

Vi vil legge til noen hindringer som Flakse kan fly igjennom.

### Sjekkliste

- ☐ Klikk på **tegn ny figur** knappen.
- ☐ Gi den nye figuren navnet rør.
- ☐ Hvis drakten er i **Punktgrafikk** klikk på **Bytt til vektorgrafikk** knappen.
- ☐ Klikk på **Zoom -** så du kan se hele tegneområdet.
- ☐ Klikk på **Rektangel**, velg en farge og klikk på **Fylt rektangel** knappen nederst til venstre.
- ☐ Klikk og dra to bokser, en fra toppen og en fr bunn i midten av tegneflaten. Det skal se sånn ut:



screenshot

- ☐ Du kan skyggelegge røra ved å klikke på **Fyll farge** og klikke på en av skyggemetodene i firkantene nede til venstre. Velg to varianter av samme farge: en for forgrunnen og en for bakgrunnen. Når du klikker på en firkant med fylleverktøyet får du en fin effekt.
- ☐ Gi figuren navnet rør.



Lagre prosjektet ditt

## Steg 5: Få røra til å bevege seg

Nå skal vi få røra til å flytte seg og gjøre plasseringen tilfeldig slik at vi får en hinderløype til Flakse.



### Sjekkliste

- ☐ Klikk på rør-figuren og velg **Skript**.
- ☐ Legg til dette skriptet:



---

## Test prosjektet

Klikk det grønne flagget: Kommer det mange rør flygende mot Flakse? Har rørene åpninger til å fly gjennom? Om du synes det er vanskelig å fly Flakse gjennom åpningene kan du endre på åpningen mellom rørene med tegneverktøyet, eller du kan lage Flakse mindre.

---

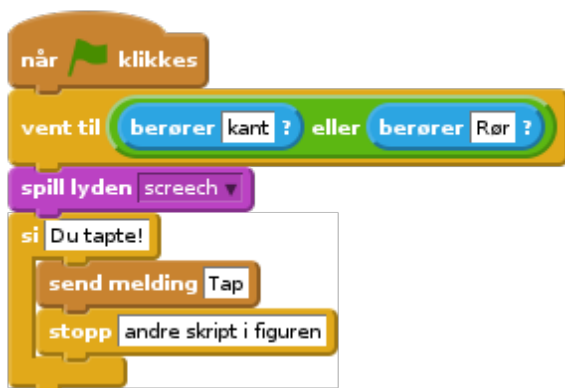
## Lagre prosjektet

# Steg 6: Finn ut om Flakse kræsjer med rørene

For at spillet skal bli vanskelig må spilleren styre Flakse gjennom åpningene mellom rørene uten å komme borti hverken rør eller kanten av skjermen. Vi skal legge til noen blokker for å merke om Flakse kræsjer.

## Sjekkliste

- ☐ Vi legger til en lyd som vi kan spille når Flakse kræsjer. Kliukk på Flakse og så på **Lyder**.
- ☐ Klikk på **Velg lyd fra biblioteket**.
- ☐ Velg en kræsje lyd for Flakse. screech er en kul lyd.
- ☐ Klikk deg tilbake til **Skript** fliken.
- ☐ Legg til dette skriptet:



☐ Klikk på Rør og legg til dette skriptet:



---

## Test prosjektet ditt

Klikk det grønne flagget: Stopper spillet når Flakse kommer borti et rør eller kanten av brettet?

---

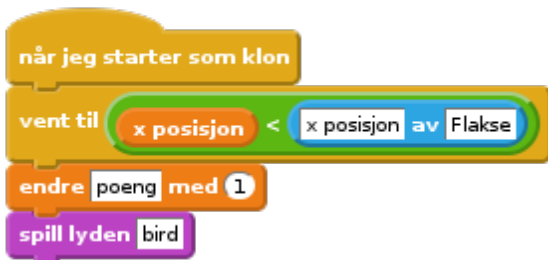
## Lagre prosjektet ditt

### Steg 7: Legg til poeng

Spilleren skal score ett poeng hver gang Flakse flyr gjennom en røråpning.

#### Sjekkliste

- ☐ Vi legger til en lyd hver gang Flakse scorer ett poeng. Klikk på Rør og legg til en lyd. bird er et lurt valg.
- ☐ Gå tilbake til Skript .
- ☐ Lag en ny variabel for alle figurer og kall den poeng.
- ☐ Legg til et skript som setter poengene til 0 når det grønne flagget klikkes.
- ☐ Legg til dette skriptet:



## Test prosjektet ditt

Klikk det grønne flagget: scorer du poeng når Flakse flyr forbi en åpning mellom rørene?



## Lagre prosjektet ditt

### Ting du kan prøve



1. Kan du lage dette spillet lettere eller vanskeligere?
2. Supert, du har laga ferdig spillet i sin enkleste form. Her er noen flere ting du kan prøve:

### Utfordring 1: legg til rekorder



Lag en ny variabel og klikk **Nett variabel (lagret på nett)** boksen. Kall variabelen **Rekord**



når spillet er ferdig må du sjekke om det er en ny rekord:



## Test prosjektet

Klikk det grønne flagget: Oppdaterer rekorden seg for hver gang du setter ny rekord?



Lagre prosjektet ditt

## Utfordring 2: legg til tyngdekraft

Når noe faller på grunn av tyngdekraft øker farten jo lenger fallet varer. Vi skal prøve å etterligne denne måten å falle på.



Legg til en ny variabel `for denne figuren` for Flakse og kall den `løft`.

Endre Flakses falleskript:



og endre Flakses flakseskript:



Test prosjektet ditt

Klikk det grønne flagget: dette Flakse forteller jo lenger han dette?





Lagre prosjektet ditt

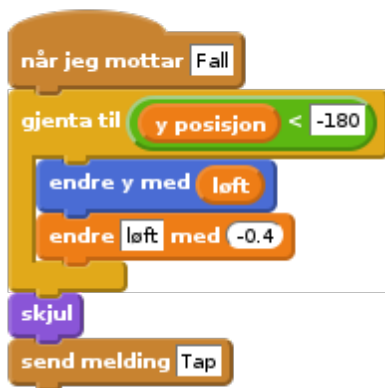
### Utfordring 3: dett ut av skjermen

Når spilleren taper skal Flakse dette ned og ut av skjermen.



Bytt ut send meldingen Tap blokka med send meldingen fall

Legg til dett eskriptet:



ikke glem å legge til en vis-blokk og sett Flakses retning når spillet starter på nytt.



Test prosjektet ditt

Klikk det grønne flagget: detter Flakse ut av skjermen når han treffer et rør? Flyr Flakse riktig vei når spillet starter igjen?



Lagre prosjektet ditt

Gratulerer, du er ferdig med spillet! Hva er rekorden din? ikke glem å dele spillet med vennene dine. Trykk på Legg ut for at andre skal få prøve!

Dette kurset er utviklet av [Code Club \(http://www.codeclub.org.uk/\)](http://www.codeclub.org.uk/) i England, og oversatt til norsk av [Kodeklubben \(http://www.kodeklubben.no/\)](http://www.kodeklubben.no/).