Verda - del 2

Skrevet av: Kine Gjerstad Eide

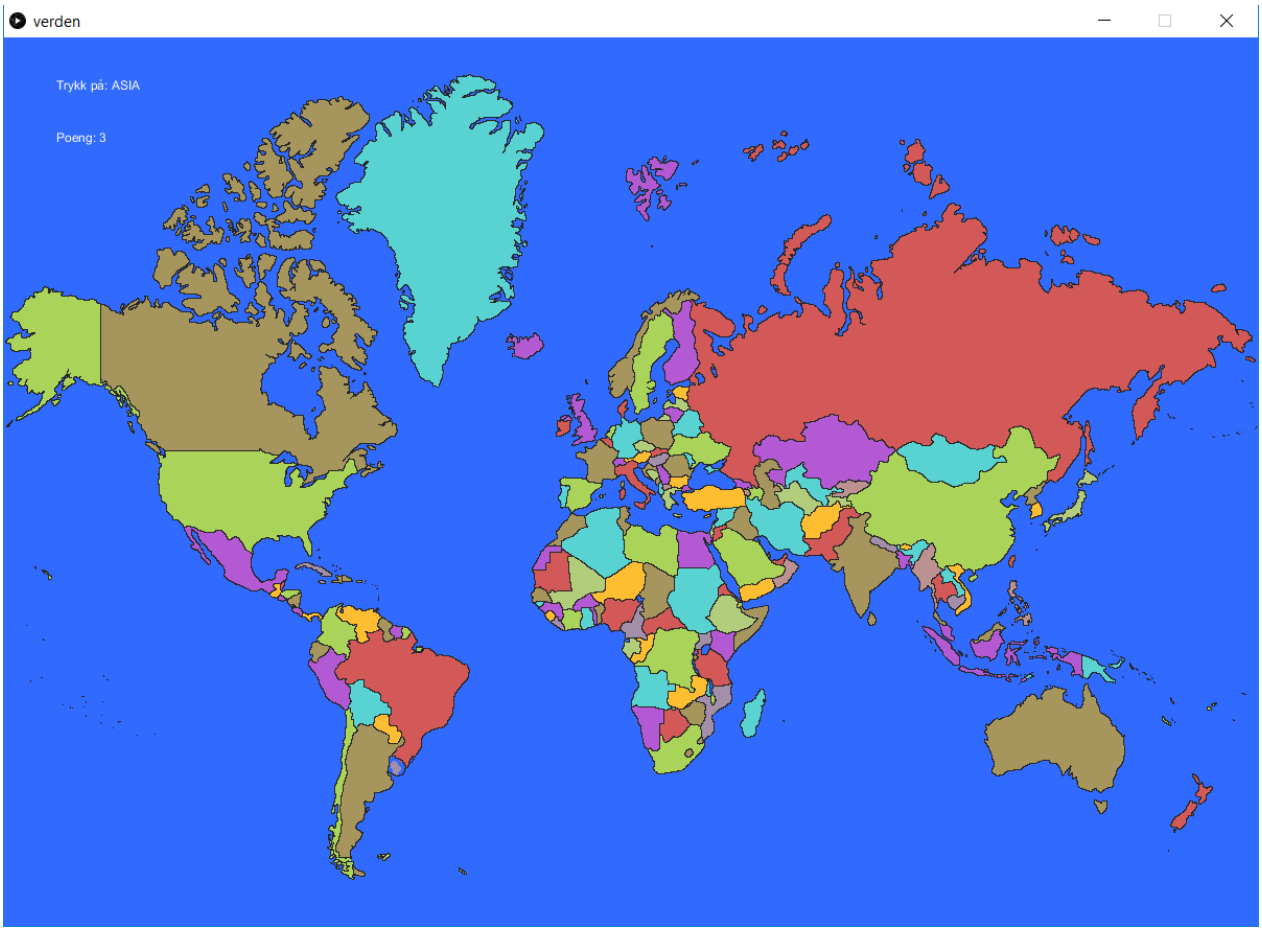
Oversatt av: Stein Olav Romslo

Kurs: Processing
Tema: Tekstbasert
Fag: Matematikk, Programmering, Samfunnsfag
Klassetrinn: 8.-10. klasse, Videregående skole

introduksjon

Denne oppgåva byggjer på oppgåva Verda (../verden_del1/verden_del1_nn.html). I denne oppgåva skal me fortsetje med verdsspelet! No skal me kome så langt på spelet at namnet på ein verdsdel skal dukke opp i vindauget, og så er målet å trykkje på den riktige verdsdelen.

Når me er ferdige skal oppgåva sjå slik ut:



Steg 0: Oppsummering frå introduksjonsoppgåva

Her er koden me laga i introduksjonsoppgåva:

```
PImage verdskartet;

void setup(){
  size(1200, 850);
  verdskartet = loadImage("world-map.png");
}
void draw(){
  image(verdskartet, 0, 0);
  text("X: " + mouseX, 50, 50);
  text("Y: " + mouseY, 50, 100);

  if(mouseX > 365){
      if(mouseX < 694){
          if(mouseY < 455){
              if(mouseY > 33){
                text("EUROPA", 950, 50);
              }
          }
      }
  }
}
```

Til no har me fått det til å dukke opp "EUROPA" på skjermen når musepeikaren blir halde over verdsdelen Europa på kartet.

Fyrst sa me at Europa er firkanta. Så fekk me koordinatane til musepeikaren til å synast som tekst i vindauget, slik at me kunne finne koordinatane til kvar grense. Så brukte me desse for å lage `if`-setningar til å sjekke med.

Me laga ei `if`-setning for kva side i firkanten som utgjorde grensa til Europa. Desse fann me ut at me måtte skrive inni kvarandre slik at me fekk sjekka om ein var innanfor alle fire grensene samstundes.

Under har me laga ei sjekkliste. Viss du får til alle punkta i lista, så er det ikkje noko problem for deg å gjere reisten av oppgåva. Viss du synest det er vanskeleg, så kan det vere lurt å gjere den førre oppgåva før du går vidare.

Sjekkliste

- [ ] Skriv inn koden som vart gjort i introduksjonsoppgåva og få den til å køyre.
- [ ] Flytt teksten som viser koordinatane ned til høgre hjørne av skjermen.
- [ ] Få teksten "EUROPA ER 10 180 000 KVADRATMETER" til å visast på skjermen i staden for berre "EUROPA" når du heldt musepeikaren over Europa.
- [ ] Endre storleiken til vindauget slik at det når frå toppen til botnen av skjermen din.

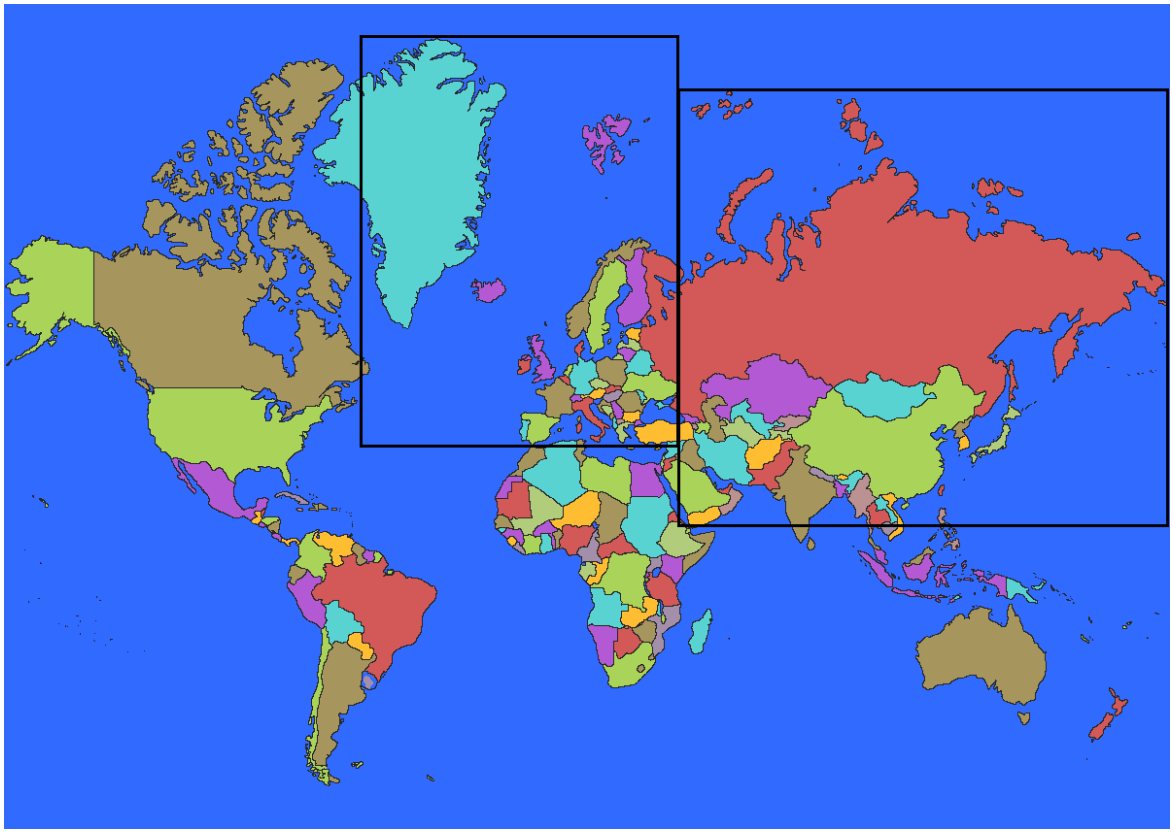
Fekk du til alle punkta? Still alt saman tilbake til koden frå sist før du går vidare.

Steg 1: Legg til Asia

I introduksjonsoppgåva lærte du korleis du la til Europa, så no skal du leggje til Asia på same måte. Me har delt det opp i mindre delar for deg. Under har me laga ei sjekkliste som du må følgje. Viss nokre av punkta er vanskelege, så kan du alltids sjekke med introduksjonsoppgåva!

Sjekkliste

- [ ] Last ned dette biletet og få det til å visast i vindauget i staden for det du har no.



- [ ] Finn ut kor grensene til Asia er og lagre dei som kommentarar i koden din. Du lagar kommentarar ved å setje to skråstrekar fyrst: `//` .

- [ ] Skriv `if` -setninga for den fyrste grensa og sjekk at det fungerer.

- [ ] Gjer eit metodekall på `text();` inni `if` -setninga der den fyrste parameteren er "ASTA". Dei to neste parametrane kan vere dei same som dei to siste i metodekallet som får "EUROPA" til å visast på skjermen.

- [ ] Skriv `if` -setninga for den andre grensa og sjekk at det fungerer. Hugs at alle `if` -setningane må stå inni kvarandre.

- [ ] Skriv `if` -setninga for den tredje grensa og sjekk at det fungerer.

- [ ] Skriv `if` -setninga for den fjerde grensa og sjekk at det fungerer.

- [ ] Bytt tilbake til biletet av verdskartet utan grensene teikna inn.

- [ ] Test at alt fungerer. No skal du sjå teksten "ASIA" i vindauget når du heldt musepeikaren over Asia, og "EUROPA" når du heldt musepeikaren over Europa.

Her er koden så langt.

```
PImage verdskartet;

void setup(){
  size(1200, 850);
  verdskartet = loadImage("world-map.png");
}
void draw(){
  image(verdskartet, 0, 0);
  text("X: " + mouseX, 50, 50);
  text("Y: " + mouseY, 50, 100);

  if(mouseX > 365){
      if(mouseX < 694){
          if(mouseY < 455){
              if(mouseY > 33){
                text("EUROPA", 950, 50);
              }
          }
      }
  }

  if(mouseX > 694){
      if(mouseX < 1197){
          if(mouseY < 537){
              if(mouseY > 88){
                text("ASIA", 950, 50);
              }
          }
      }
  }
}
```

Steg 2: Slå saman if-setningar

No som du har lært korleis du set opp `if` -setningar inni kvarandre, så er det på tide å lære korleis du kan slå dei saman. Når du set `if` -setningar inni kvarandre kan du lese dei som dette:

```
Viss musepeikaren er innanfor venstre grense
    og viss musepeikaren er innanfor høgre grense
        og viss musepeikaren er innanfor nedre grense
            og viss musepeikaren er innanfor øvre grense,
                så skal me skrive "EUROPA" på skjermen.
```

Gjer me det om til litt meir vanleg tekst, så ville me sagt:

```
Viss musepeikaren er innanfor venstre grense og musepeikaren er innanfo
r høgre
  grense og musepeikaren er innanfor nedre grense og musepeikaren er in
nanfor
  øvre grense,
    så skal me skrive "EUROPA" på skjermen.
```

Dette kan me gjere når me kodar òg. La oss setje inn det me har kode så langt i setninga over. Då får me dette:

```
if(mouseX > 365 og mouseX < 694 og mouseY < 455 og mouseY > 33)]{
    text("EUROPA", 950, 50);
}
```

Til no har me ikkje lagt til noko nytt i koden. Me har berre sett alle testane på same linje med "og" mellom. Me kan ikkje bruke "og" i Processing, men me kan bruke `&&`, og det tyder akkurat det same! La oss setje inn `&&` for "og". Då får me:

```
if(mouseX > 365 && mouseX < 694 && mouseY < 455 && mouseY > 33)]{
    text("EUROPA", 950, 50);
}
```

Det kan hende du synest denne `if`-setninga er litt lang og vanskeleg å lese, men no har me spart mykje plass i koden.

Sjekkliste

- [ ] Slå saman dei fire `if`-setningane som sjekkar grensene til Europa til å vere berre ei `if`-setning.
- [ ] Sjekk at koden fungerer.
- [ ] Skriv om `if`-setningane for Asia, slik at du sjekkar alle grensene i ei `if`-setning.

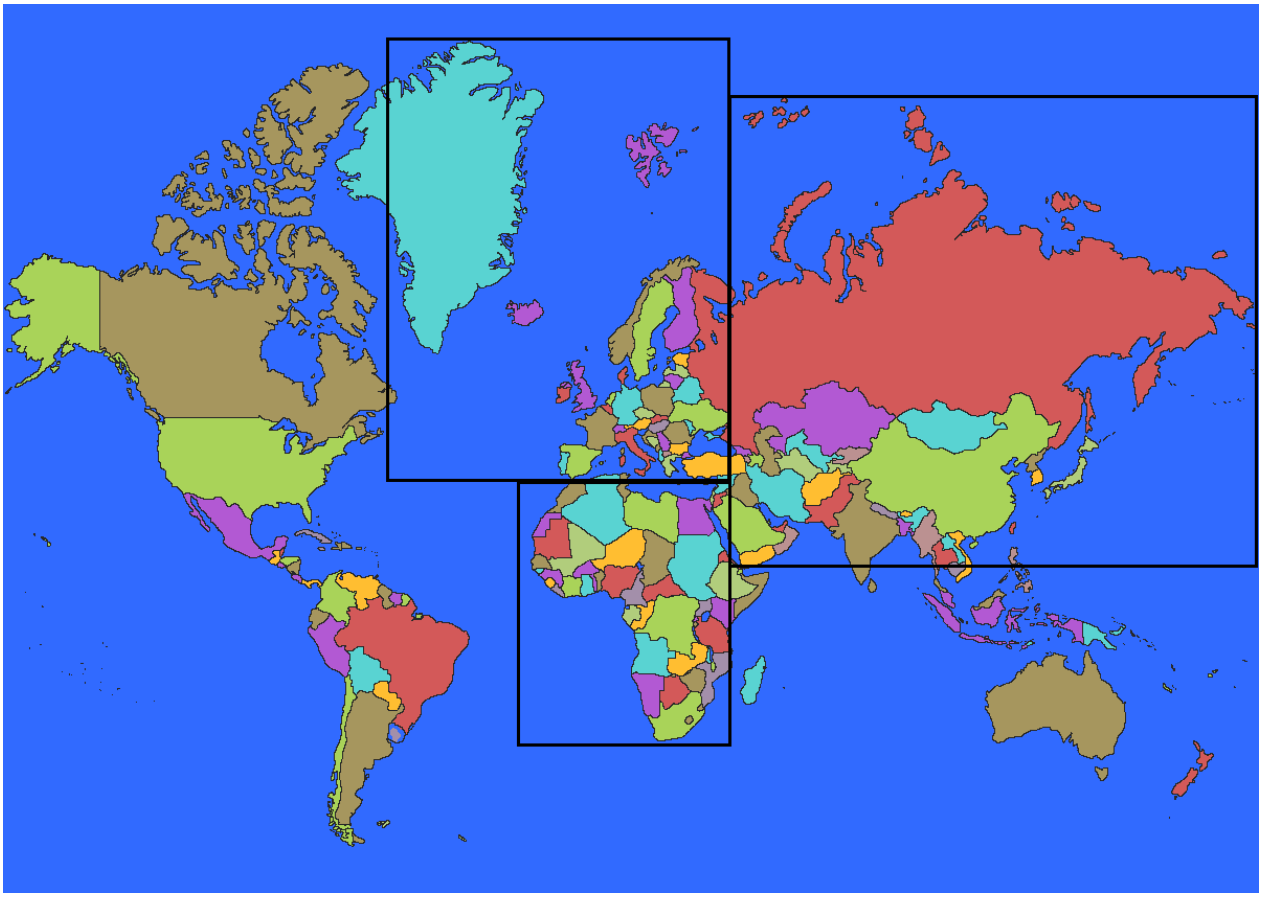
- [ ] Sjekk at koden fungerer.

Her er koden så langt.

```
PImage verdskartet;

void setup(){
  size(1200, 850);
  verdskartet = loadImage("world-map.png");
}
void draw(){
  image(verdskartet, 0, 0);
  text("X: " + mouseX, 50, 50);
  text("Y: " + mouseY, 50, 100);

  if(mouseX > 365 && mouseX < 694 && mouseY < 455 && mouseY > 33){
                text("EUROPA", 950, 50);
  }

  if(mouseX > 694) && mouseX < 1197 && mouseY < 537 && mouseY > 88){
                text("ASIA", 950, 50);
  }
}
```

Steg 3: Sjekk Afrika med to if-setningar

No skal me leggje til Afrika med ei `if` -setning på same måte. No kan du velje om du vil følgje sjekklista under, eller om du vil skrive den heilt åleine. Her er eit bilete med grensa til Afrika:



Sjekkliste

- [ ] Bytt ut verdskartbiletet med biletet over.
- [ ] Finn koordinatane på de fire grensene til Afrika og skriv dei inn som kommentarar i koden din.
- [ ] Kall på `text();` og bruk "AFRIKA" som fyrste parameter.
- [ ] Skriv koden til den fyrste grensa.
- [ ] Sjekk at koden fungerer og at `if` -setninga gjer det den skal.
- [ ] Skriv `&&` og legg til testen for den andre grensa til Afrika i same `if` -setning. Munnleg blir det

```
if(er musepeikaren innanfor den fyrste grensa && er musepeikaren innanf
or den
  andre grensa){
    vis teksten "AFRIKA" i vindauget.
}
```

- [ ] Test at koden fungerer som den skal.

- [ ] Skriv `&&` og legg til testen for den tredje grensa til Afrika i same `if` -setning.

- [ ] Test at det fungerer som det skal.

- [ ] Skriv `&&` og legg til testen for den fjerde grensa til Afrika i same `if` -setning.

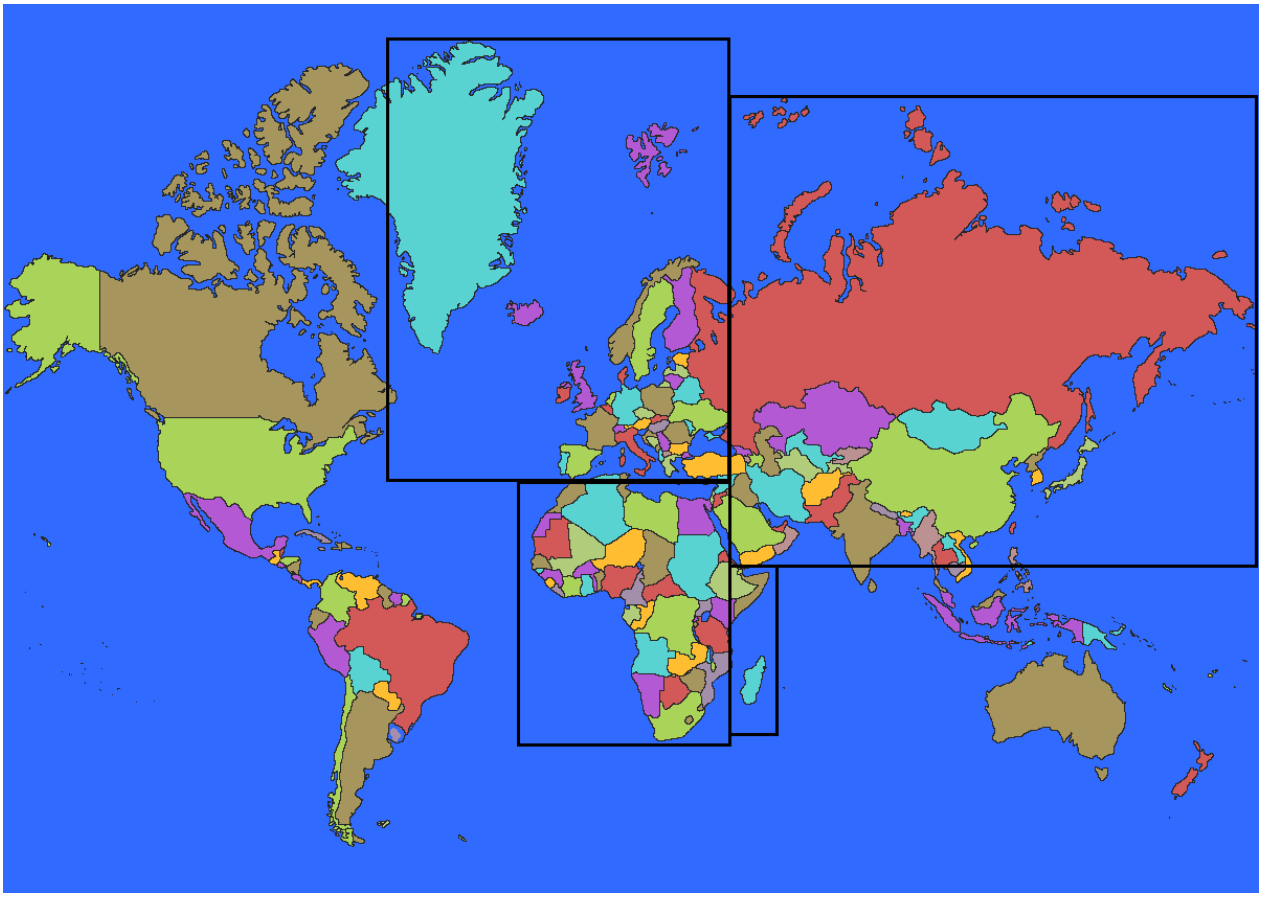
- [ ] Test at koden fungerer og at "AFRIKA" visast i vindauget når du heldt musepeikaren over Afrika.

No som Afrika fungerer, så skal me legje til ei lita rute til. Som du ser på kartet er ikkje Madagaskar, Somalia, halve Etiopia og Djibouti med i den fyrste firkanten. Så når me heldt musepeikaren over desse landa, så kjem ikkje teksten "AFRIKA" opp på skjermen. Det må me fikse.

Me har laga ein ny firkant som berre inkluderer desse landa. No har du skrive så mange `if` -setningar du dette klarar du utan sjekkliste. Viss det er vanskeleg kan du bruke sjekklista over, for du skal gjere akkurat det same, men med andre koordinatar.

Sjekkliste

- [ ] Lag ei ny `if` -setning som sjekkar grensene til den nye firkanten. Her er biletet:



Her er koden vår så langt. No er det ein del `if` -setningar!

```
PImage verdskartet;

void setup(){
  size(1200, 850);
  verdskartet = loadImage("world-map.png");
}
void draw(){
  image(verdskartet, 0, 0);
  text("X: " + mouseX, 50, 50);
  text("Y: " + mouseY, 50, 100);

  if(mouseX > 365 && mouseX < 694 && mouseY < 455 && mouseY > 33){
                text("EUROPA", 950, 50);
  }

  if(mouseX > 694) && mouseX < 1197 && mouseY < 537 && mouseY > 88){
                text("ASIA", 950, 50);
  }

  if(mouseX > 493) && mouseX < 695 && mouseY < 708 && mouseY > 455){
                text("AFRIKA", 950, 50);
  }

  if(mouseX > 694) && mouseX < 740 && mouseY < 698 && mouseY > 537){
                text("AFRIKA", 950, 50);
  }
}
```

Steg 4: Lag ein metode i staden for alle if-setningane

Me skal forenkle koden vår endå meir. Sidan me skal bruke så mange `if` -setningar vil me spare oss ein del arbeid. Du ser at dei er nesten heilt like, det er berre grenseverdiane som er ulike. Når me ser at me skal skrive mykje veldig lik kode på denne måten, så kan me ofte spare tid på å setje det som er likt inn i ein metode, og så bruke eit metodekall i staden for å skrive same kode om att og om att.

Me har skrive metodane `setup` og `draw` , medan `text();` er ein metode me har kalla på. Når du trykkar play blir det gjort eit kall på `setup` , det skjer berre ein gong. Så kallast `draw` på nytt og på nytt heilt til programmet blir stoppa.

Me skal skrive ein ny metode, den plasserar me under `draw` . Me har valt å gi metoden namnet `sjekkVerdsdel` . Då blir det slik:

```
void sjekkVerdsdel(){

}
```

Inne i krøllparentesane skal me skrive den koden me vil gjenta, altså dei lange `if`-setningane.

La oss starte med å setje inn ei av `if`-setningane. Me har valt å kopiere inn `if`-setninga som sjekkar om musepeikaren er innanfor grensene til Europa, så no ser metoden slik ut:

```
void sjekkVerdsdel(){
    if(mouseX > 365 && mouseX < 694 && mouseY < 455 && mouseY > 33)]{
        text("EUROPA", 950, 50);
    }
}
```

Me har ikkje så lyst å skrive inn ei `if`-setning for kvar verdsdel i den nye metoden, då er me jo like langt. Men for å berre ha ei `if`-sentning, så byttar me ut alle dei tala som varierer frå verdsdel til verdsdel med variablar! Me gir variabelnamn som er beskrivande slik at koden blir enklare å lese seinare. Ein variabel kan endrast når programmet køyrer, men eit tal som er skrive inn direkte kan ikkje endrast. I den fyrste testen sjekkar me om musepeikaren er innanfor den fyrste grensa, altså grensa til høgre. Sidan me ikkje kan bruke `æ`, `ø` eller `å` skriv me `høgre` med `o`, altså `hogreGrense`. Vi set inn denne variabelen for talet `365` og får:

```
void sjekkVerdsdel(){
    if(mouseX > hogreGrense && mouseX < 694 && mouseY < 455 && mouseY >
33)]{
        text("EUROPA", 950, 50);
    }
}
```

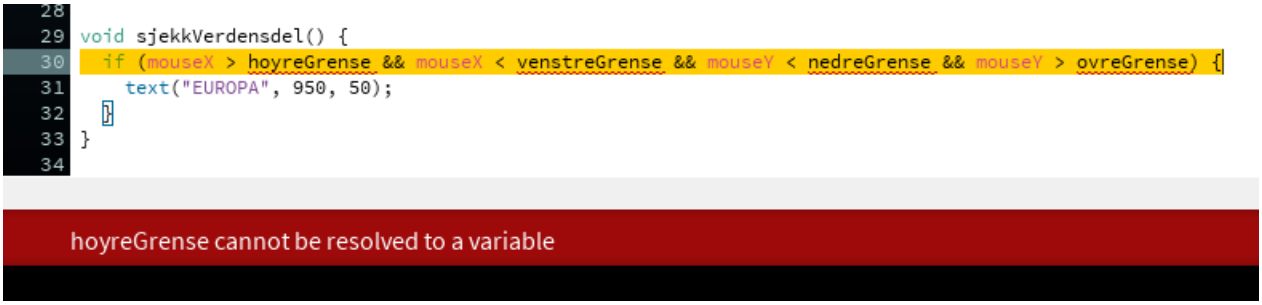
Me gjer det same med dei andre tre grensene.

Sjekkliste

- [ ] Finn passande namn til dei andre grensene, og bytt ut tala i `if`-setninga med dei namna.

- [ ] Køyr programmet. Kva skjer?

No fekk me ei feilmelding som ser slik ut:



Programmet seier at `hogreGrense` ikkje kan bli ein variabel. Det er fordi me ikkje har skrive kode som fortel programmet at me vil bruke `hogreGrense` som ein variabel fyrst. Førebels trur programmet at me har gjort ein skrivefeil. Programmet veit at me skal samanlikne musepeikaren sin koordinat med eitt eller anna tal, men akkurat no er ikkje `hogreGrense` noko tal, det er berre nokre bokstavar.

Me må gjere om `hogreGrense` til ein variabel av typen `int`, altså heiltal. Det heiter å *deklarare* vairabelen. Me kan gjere det fleire stader, både inni den nye metoden, utanfor alle metodane eller inni parentesane til metoden. Kor me vel å gjere det er avhengig av korleis me skal bruke metoden.

Sidan me vil sjekke ulike grenser kvar gong me kallar metoden, så er det lurt å deklarere variabelen `hogreGrense` inni parentesane til den nye metoden. Når deklarasjonen er plassert her, så må me leggje ved grenseverdien som parameter kvar gong me kallar på metoden.

Metoden ser slik ut:

```
void sjekkVerdsdel(int hogreGrense, int ...., int ...., int .... ){
    if(mouseX > hogreGrense && mouseX < .... && mouseY < .... && mouseY
> ....)]{
        text("EUROPA", 950, 50);
    }
}
```

I staden for punktum må du setje inn namna du har valt for dei ulike grensene.

Sjekkliste

- [ ] Deklarer alle dei fire variablane dine inni parentesen til den nye metoden din.
- [ ] Sjekk om programmet køyrer.

Me manglar framleis ein ting - namnet på verdsdelen. Alle verdsdelane heiter jo ikkje "EUROPA". Då må me gjere denne variabelen til ein `String`, altså ein tekst.

Sjekkliste

- [ ] Bytt ut den fyrste parameteren i metodekallet på `text();` i `if`-setninga i den nye metoden, og skriv i staden `verdsdel` (du kan velje eit anna namn om du vil det).
- [ ] Køyr programmet. Kva skjer?
- [ ] Deklarer `verdsdel` til å vere ein `String` inni parentesane til den nye metoden.
- [ ] Køyr programmet. Kva skjer?

No har me laga ein ny metode, men me har ikkje kalla på den nye metoden nokon stad. Førebels køyrer programmet vårt ein gong gjennom `setup`, så `draw` om att og om att, utan å gå til den nye metoden.

Her er koden så langt. Hugs at me sikkert har litt ulike variabelnamn for dei ulike grensene. Så lenge programmet køyrer er det ikkje noko problem. Det kan vere greitt om deklarasjonane på dei ulike grensene og namnet på verdsdelen ligg i same rekkefølgje, men det er berre for at det skal bli enklare å feilsøke i koden seinare.

```
PImage verdskartet;

void setup(){
  size(1200, 850);
  verdskartet = loadImage("world-map.png");
}
void draw(){
  image(verdskartet, 0, 0);
  text("X: " + mouseX, 50, 50);
  text("Y: " + mouseY, 50, 100);

  if(mouseX > 365 && mouseX < 694 && mouseY < 455 && mouseY > 33){
                text("EUROPA", 950, 50);
  }

  if(mouseX > 694) && mouseX < 1197 && mouseY < 537 && mouseY > 88){
                text("ASIA", 950, 50);
  }

  if(mouseX > 493) && mouseX < 695 && mouseY < 708 && mouseY > 455){
                text("AFRIKA", 950, 50);
  }

  if(mouseX > 694) && mouseX < 740 && mouseY < 698 && mouseY > 537){
                text("AFRIKA", 950, 50);
  }
}

void sjekkVerdsdel(String verdsdel, int hogreGrense, int venstreGrense,
  int nedreGrense, int ovreGrense){
    if(mouseX > hogreGrense && mouseX < venstreGrense &&
      mouseY < nedreGrense && mouseY > ovreGrense)]{
        text(verdsdel, 950, 50);
    }
}
```

Steg 5: Bruk den nye metoden

For å bruke den nye metoden må me gjere eit metodekall på den. På same måte som me har gjort eit metodekall på `text();` og `size();`, så skal me no gjere eit metodekall på `sjekkVerdsdel();`. Dette gjer me inne i `draw`-metoden.

Sjekkliste

- [ ] Gjer eit metodekall på `sjekkVerdsdel();` inne i `draw`-metoden.

- [ ] Køyr koden. Kva skjer? Forstår du kva feilmeldinga tyder?

```
27
28    sjekkVerdensdel();
29  }
30
31  void sjekkVerdensdel(String verdensdel, int hoyreGrense, int venstreGrense, int nedreGrense, int ovreGrense) {
32    if (mouseX > hoyreGrense && mouseX < venstreGrense && mouseY < nedreGrense && mouseY > ovreGrense) {
33      text(verdensdel, 950, 50);
34    }
35  }
36
37
```

```
The method sjekkVerdensdel(String, int, int, int, int) in the type verden is not applicable for the arguments ()
```

- [ ] Gjer om metodekallet til ein kommentar ved å setje to skråstrekar fyrst på linja.

- [ ] Ta bort parametrane i eitt av metodekalla på `text();`.

- [ ] Køyr programmet. Kva skjer? Forstår du feilmeldinga?

```
7   void draw() {
8     image(verdenskartet, 0, 0);
9     text();
10    text("Y: " + mouseY, 50, 100);
11
12    if (mouseX > 365 && mouseX < 694 && mouseY < 455 && mouseY > 33) {
13      text("EUROPA", 950, 50);
14    }
15
16    if (mouseX > 694 && mouseX < 1197 && mouseY < 537 && mouseY > 88) {
17      text("ASIA", 950, 50);
18    }
19
20    if (mouseX > 493 && mouseX < 695 && mouseY < 708 && mouseY > 455) {
21      text("AFRIKA", 950, 50);
22    }
23
24    if (mouseX > 694 && mouseX < 740 && mouseY < 698 && mouseY > 537) {
25      text("AFRIKA", 950, 50);
26    }
27
28    //sjekkVerdensdel();
29  }
30
31  void sjekkVerdensdel(String verdensdel, int hoyreGrense, int venstreGrense, int nedreGrense, int ovreGrense) {
32    if (mouseX > hoyreGrense && mouseX < venstreGrense && mouseY < nedreGrense && mouseY > ovreGrense) {
33      text(verdensdel, 950, 50);
34    }
35  }
36
37
38
```

```
The method text(char, float, float) in the type PApplet is not applicable for the arguments ()
```

Det manglar rett og slett parametrar i metodekalla. For metodekallet på `text();` treng me fyrst ein `char` og så to `float`. Det tyder at me fyrst treng teksten som skal visast i vindauget, og så to tal som bestemmer kor teksten skal visast.

For metodekallet på `sjekkVerdsdel();` så treng me fyrst ein `String` som seier kva verdsdel me sjekkar om musepeikaren er i, så fire `int`, som fortel kor grensene til verdsdelen er.

Me må setje inn parametrar i metodekallet.

Sjekkliste

- [ ] Set inn parametrane i metodekallet på `text();` att.

- [ ] Start med Europa. Den fyrste parameteren skal vere `"EUROPA"`. Hugs hermeteikna, elles veit ikkje programmet at det er ein tekst. Skil parametrane frå kvarandre med komma. Den neste parameteren er den fyrste grensa, altså `365`. Så kjem den andre grensa, altså den på venstre side av firkanten me brukte, som er `694`. Så er det den nesre grensa, `455`, og til slutt den øvre grensa, som me fann ut at er `33`.

- [ ] Køyr programmet. Kva skjer?

- [ ] No vil "EUROPA" visast dobbelt opp på skjermen, sidan me både sjekkar om musepeikaren er over Europa i `if`-setninga i `draw`-metoden og så i det nye metodekallet vårt. Ta bort `if`-setninga som er skrive i `draw` og som sjekkar Europa.

- [ ] Køyr programmet på nytt. Fungerer det framleis som det skal?

Her er eit bilete av koden. Det er viktig at du har fjerna riktig `if`-setning og at du har brukt riktige parametrar i metodekallet på `sjekkVerdsdel();`.

```
PImage verdskartet;

void setup(){
  size(1200, 850);
  verdskartet = loadImage("world-map.png");
}
void draw(){
  image(verdskartet, 0, 0);
  text("X: " + mouseX, 50, 50);
  text("Y: " + mouseY, 50, 100);

  if(mouseX > 694) && mouseX < 1197 && mouseY < 537 && mouseY > 88){
                text("ASIA", 950, 50);
  }

  if(mouseX > 493) && mouseX < 695 && mouseY < 708 && mouseY > 455){
                text("AFRIKA", 950, 50);
  }

  if(mouseX > 694) && mouseX < 740 && mouseY < 698 && mouseY > 537){
                text("AFRIKA", 950, 50);
  }

  sjekkVerdsdel("EUROPA", 365, 694, 455, 33);
}

void sjekkVerdsdel(String verdsdel, int hogreGrense, int venstreGrense,
  int nedreGrense, int ovreGrense){
    if(mouseX > hogreGrense && mouseX < venstreGrense &&
      mouseY < nedreGrense && mouseY > ovreGrense)]{
        text(verdsdel, 950, 50);
    }
}
```

Steg 6: Bruk `sjekkVerdsdel`-metoden til alle verdsdelane

No skal du bruke metoden `sjekkVerdsdel();` til å sjekke Asia og Afrika. Sidan du allereie har funne koordinatane på grensene, så treng du ikkje gjere det på nytt.

Sjekkliste

- [ ] Gjer eit metodekall på `sjekkVerdsdel();` der parametrane er dei du har funne for Asia.

- [ ] Fjern `if` -setninga som sjekkar om musepeikaren er over Asia.

- [ ] Køyr koden og sjekk at "ASIA" kjem opp når du heldt musepeikaren innanfor grensene til Asia.

- [ ] Gjer to metodekall på `sjekkVerdsdel();` og få den til å sjekke om musepeikaren er over Afrika.

- [ ] Fjern dei to `if` -setningane som sjekkar om musepeikaren er over Afrika.

- [ ] Sjekk at koden fungerer.

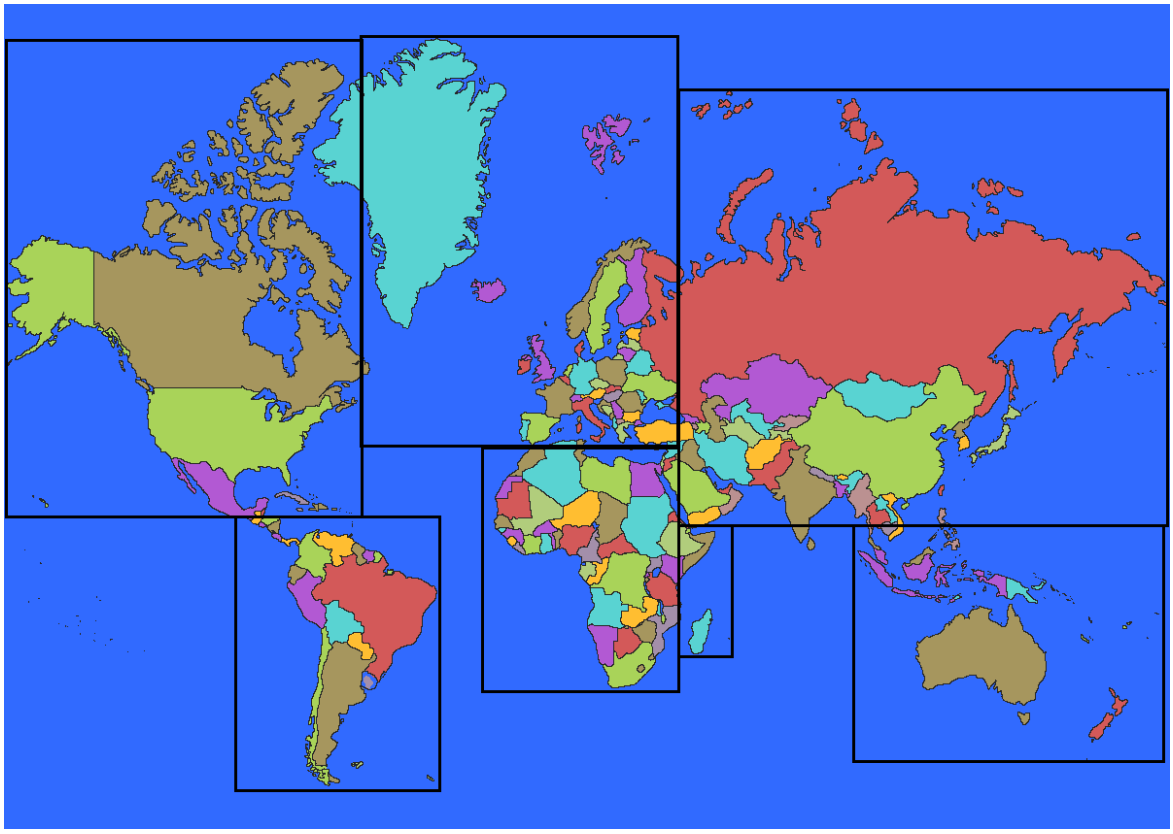
Her er koden vår så langt.

```
PImage verdskartet;

void setup(){
  size(1200, 850);
  verdskartet = loadImage("world-map.png");
}
void draw(){
  image(verdskartet, 0, 0);
  text("X: " + mouseX, 50, 50);
  text("Y: " + mouseY, 50, 100);

  sjekkVerdsdel("EUROPA", 365, 694, 455, 33);

  sjekkVerdsdel("ASIA", 694, 1197, 537, 88);

  sjekkVerdsdel("AFRIKA", 493, 695, 708, 455);

  sjekkVerdsdel("AFRIKA", 694, 740, 698, 537);
}

void sjekkVerdsdel(String verdsdel, int hogreGrense, int venstreGrense,
  int nedreGrense, int ovreGrense){
    if(mouseX > hogreGrense && mouseX < venstreGrense &&
      mouseY < nedreGrense && mouseY > ovreGrense)]{
        text(verdsdel, 950, 50);
    }
}
```

No må me leggje til dei siste verdsdelane. Her har me eit kart der me viser alle saman. No skal me gjere metodekall på `sjekkVerdsdel();` og leggje til grenseverdiane til Nord-Amerika, Sør-Amerika og Oseania. Kartet vårt manglar verdsdelen Antarktis (Sørpolen), men det kan du ordne ved å velje eit anna kart.

Sjekkliste

- [ ] Bytt ut slik at dette biletet visast:



- [ ] Gjer eit metodekall på `sjekkVerdsdel();`.

- [ ] Set inn den fyrste parameteren til metoden, som er `"OSEANIA"`.

- [ ] Finn grenseverdiane til Oseania og skriv desse rett inn som parametrar i metodekallet i staden for å skrive dei som kommentarar i koden fyrst. Start med den venstre grensa, så den høgre, den nedre og til slutt den øvre.

- [ ] Sjekk at koden fungerer. No skal det stå "OSEANIA" på skjermen når du heldt musepeikaren over Oseania. Viss det ikkje fungerte som det skal, så kan du prøve å skrive grensene som kommentarar fyrst, slik at du er heilt sikker på kva tal som skal stå kor. Sjekk og at rekkefølgja på parametrane vart riktig.

- [ ] Gjer eit nytt metodekall på `sjekkVerdsdel();` og set inn parametrane for Nord-Amerika. Sidan du akkurat har gjort det for Oseania, så er det berre å gjere det same her.

- [ ] Test at koden fungerer. No skal namnet på alle verdsdelane bortsett frå Sør-Amerika dukke opp på skjermen når du heldt musepeikaren over dei.

- [ ] Skriv kode slik at "SØR-AMERIKA" visast på skjermen når du heldt musepeikaren over verdsdelen.

- [ ] Test at koden fungerer.

Her er koden vår så langt.

```
PImage verdskartet;

void setup(){
  size(1200, 850);
  verdskartet = loadImage("world-map.png");
}
void draw(){
  image(verdskartet, 0, 0);
  text("X: " + mouseX, 50, 50);
  text("Y: " + mouseY, 50, 100);

  sjekkVerdsdel("EUROPA", 365, 694, 455, 33);

  sjekkVerdsdel("ASIA", 694, 1197, 537, 88);

  sjekkVerdsdel("AFRIKA", 493, 695, 708, 455);

  sjekkVerdsdel("AFRIKA", 694, 740, 698, 537);

  sjekkVerdsdel("OSEANIA", 874, 1194, 780, 537);

  sjekkVerdsdel("NORD-AMERIKA", 2, 365, 528, 36);

  sjekkVerdsdel("SØR-AMERIKA", 238, 449, 810, 528);
}

void sjekkVerdsdel(String verdsdel, int hogreGrense, int venstreGrense,
  int nedreGrense, int ovreGrense){
    if(mouseX > hogreGrense && mouseX < venstreGrense &&
      mouseY < nedreGrense && mouseY > ovreGrense)]{
        text(verdsdel, 950, 50);
    }
}
```

Steg 7: Trekk ein tilfeldig verdsdel

No har me fått alle verdsdelane til å visast når me heldt musepeikaren over dei. Då er det på tide å gjere dette programmet om til eit spel! Me vil at namnet på ein tilfeldig verdsdel skal dukke opp på skjermen, og så skal me få poeng om me klarar å trykke på rett verdsdel.

For å få dette til å skje treng me ei liste med alle verdsdelane, og så kode for å trekkje tilfeldig frå lista.

Me startar med lista. Den skal innehalde orda "AFRIKA", "ASIA", "EUROPA", "SØR-AMERIKA", "NORD-AMERIKA" og "OSEANIA". Dette er tekst, så då brukar me ei liste for å lage tekst. Då me oppretta metoden `sjekkVerdsdel` brukte me `String` inni parentesen for å ta imot ein tekst. Då var det berre eitt ord, ein tekst, men no treng me noko som kan innehalde mange separate tekstar. Det er ganske enkelt, me må berre setje `[]` bak `String` og så veit programmet at det er ei liste med tekstar me skal ha. Me veit nøyaktig kva lista skal innehalde, så me kan fylle den opp samstundes som me deklarerer den.

Me deklarerer lista vår heilt i toppen av programmet, utanfor alle metodane. Me kan gi lista det namnet me vil. Her har me valt `alleVerdsdelane`, så kodelinja ser slik ut:

```
String[] alleVerdsdelane = {"ASIA", "EUROPA", "NORD-AMERIKA",
 "SØR-AMERIKA", "OSEANIA", "AFRIKA"};
```

Der er alt me treng for å lage lista. Denne typen liste heiter `String array`.

Sjekkliste

- [ ] Deklarer eit `String array`. Du kan gi det akkurat det namnet du vil, men me anbefalar `alleVerdsdelane`, slik at det er lett å samanlikne koden din med vår.

- [ ] Køyr programmet og sjå at det køyrer som det skal.

Metoden `random();` kan kallast for å få eit tilfeldig tal. Ulikt `text();` og `size();`, så gjer ikkje `random();` noko med vindauget vårt. Det som skjer når me brukar random-metoden er at me får tilbake eit tal, og det må me setje inn i ein variabel slik at me kan bruke det.

Fyrst vil me deklarere ein variabel. Sidan `random();` gir oss eit tal av typen `float` burde variabelen eigentleg vere av typen `float`. Men når me skal bruke talet seinare må det vere av typen `int`, difor skal me gjere verdien om til `int`. Det kallast å `caste`

verdien. Namnet kan me bestemme sjølv, me har valt `tilfeldigTal`. Deklarasjonen må skje øvst i koden, anten rett over eller rett under deklarasjonen av lista. Kodelinja blir slik:

```
int tilfeldigTal;;
```

No må me faktisk få `tilfeldigTal` til å innehalde eit tilfeldig tal. Det gjer me inne i `setup`-metoden. Me startar med "munnleg kode":

```
    variabelen tilfeldigTal skal vere eit tilfeldig tal;
```

No brukar me `random();` med casting til `int`. Då blir det:

```
    tilfeldigTal = (int)random(6);
```

Parameteren til `random();` fortel kva tal det skal veljast tilfeldig mellom. Me har 6 verdsdelar i lista, så me vil at `random();` skal gi oss eit tal mellom 0 og 6. Difor skriv me 6 som parameter.

Sjekkliste

- [ ] Deklarer ein variabel av typen `int`. Gi variabelen eit passande namn, til dømes `tilfeldigTal`.
- [ ] Sjekk at koden køyrer.
- [ ] Set variabelen til å bli eit tilfeldig tal mellom 0 og 6 ved å gjere eit metodekall på `random();` og cast `int`.
- [ ] Sjekk at koden køyrer.
- [ ] Gjer eit metodekall på `text();` der fyrste parameter er `"Variabelen tilfeldigTal er: " + tilfeldigTal`. Dei to neste parametrane bestemmer du sjølv. No skal me sjekke kva tal `tilfeldigTal` har blitt til.
- [ ] Køyr programmet nokre gonger og sjekk av variabelen `tilfeldigTal` blir ulike tal.

- [ ] Gjer eit nytt metodekall på `text();` der den fyrste parameteren er `"Trykk på: " + alleVerdsdelane[tilfeldigTal]`. Dei to neste parametrane bestemmer du sjølv. Hugs at `alleVerdsdelane` er lista med verdsdelar, så det me gjer her er å seie at me skal vise verdsdelen som er på plassen til det talet som variabelen `tilfeldigTal` inneheldt.

- [ ] Sjekk at koden køyrer. Start programmet fleire gonger og sjekk at du får ulike verdsdelar.

Når me skal velje noko frå eit string array, så må me skrive namnet på arrayet og `[]`. Alt som står i lista er nummerert i den rekkefølgja me har skrive i lista. Inne i `[]` skriv me nummeret på den tingen i lista me vil ha. Nummeret må vere av typen `int`. Dette er grunnen til at me castar til `int` når me gjer metodekall på `random();`. Sjekk at tala stemmer ved å samanlikne talet som `tilfeldigTal` er med kva plass i lista verdsdelen som visast på skjermen er.

Før me går vidare skal me fjerne litt unødvendig kode.

Sjekkliste

- [ ] Fjern dei to metodekalla på `text();` som viser oss kva musepeikaren sin X- og Y-koordinat er.

- [ ] Fjern metodekallet på `text();` som har skrive i `if`-setninga i `sjekkVerdsdelar`-metoden. I neste steg set me inn ein heilt annan kode her.

- [ ] Fjern metodekallet på `text()` som viser kva tal `tilfeldigTal` er.

Her er koden så langt:

```
PImage verdskartet;
String[] alleVerdsdelane = {"ASIA", "EUROPA", "NORD-AMERIKA",
  "SØR-AMERIKA", "OSEANIA", "AFRIKA"};
int tilfeldigTal;

void setup(){
  size(1200, 850);
  verdskartet = loadImage("world-map.png");
  tilfeldigTal = (int)random(6);
}
void draw(){
  image(verdskartet, 0, 0);
  text("Trykk på: " + alleVerdsdelane[tilfeldigTal], 50, 50);

  sjekkVerdsdel("EUROPA", 365, 694, 455, 33);
  sjekkVerdsdel("ASIA", 694, 1197, 537, 88);
  sjekkVerdsdel("AFRIKA", 493, 695, 708, 455);
  sjekkVerdsdel("AFRIKA", 694, 740, 698, 537);
  sjekkVerdsdel("OSEANIA", 874, 1194, 780, 537);
  sjekkVerdsdel("NORD-AMERIKA", 2, 365, 528, 36);
  sjekkVerdsdel("SØR-AMERIKA", 238, 449, 810, 528);
}

void sjekkVerdsdel(String verdsdel, int hogreGrense, int venstreGrense,
  int nedreGrense, int ovreGrense){
    if(mouseX > hogreGrense && mouseX < venstreGrense &&
      mouseY < nedreGrense && mouseY > ovreGrense)]{
    }
}
```

Steg 8: Sjekk om spelaren trykkar på riktig verdsdel

No skal me sjekke om spelaren trykkar på riktig verdsdel. Då startar me med å sjekke om spelaren trykkar på knappen på musa.

På same måte som me kan bruke `mouseX` og `mouseY`, så kan me bruke `mousePressed`. Skilnaden er at `mouseX` og `mouseY` blir oppdatert med tal som fortel kor i vindauget musepeikaren er, medan `mousePressed` er ein boolsk variabel - altså er den anten `true` eller `false` (sann eller usann) - så me kan bruke den direkte i ei `if`-setning.

Munnleg er det dette me skal gjere:

```
Viss musepeikaren er klikka på,
    så skal me sjekke om musepeikaren er over ein og ein verdsdel
```

Det å sjekke om musepeikaren er over ein og ein av verdsdelane har me allereie gjort i `draw`-metoden ved å kalle på `sjekkVerdsdel();` mange gonger. Då må me berre setje desse inni ei `if`-setning.

Sjekkliste

- [ ] Skriv ei `if`-setning der du sjekkar om museknappen er trykka på. No skal det berre stå `mousePressed` i parentesen til `if`-setninga.
- [ ] Plasser alle metodekalla `sjekkVerdsdel();` inni `if`-setninga.
- [ ] Sjekk at koden køyrer. Det skal ikkje skje noko nytt.

Neste steg er å sjekke om spelaren har trykka på riktig verdsdel. Det gjer me i `if`-setninga som er inni metoden `sjekkVerdsdel`. No skal me sjekke om verdsdelen frå strengen som blir sendt inn til metoden er lik den verdsdelen som er trekt tilfeldig. Munnleg blir det:

```
Viss verdsdelen som er trekt tilfeldig er lik verdelsdelen frå Stringe
n,
    så visar me teksten "RIKTIG" i vindauget.
```

Me set inn det me har, då blir `if`-setninga slik:

```
if(alleVerdsdelane[tilfeldigTal] er lik verdsdel){
    text("RIKTIG", 950, 50);
}
```

No skal me sjekke om ein tekst er lik ein annan tekst. Då må me bruke ein metode som høyrer til klassa `String`. Metoden heiter `equals();` og parameteren er ordet me skal sjekke. Når me skal bruke ein metode som høyrer til ei klasse treng me berre å setje punktum mellom. Koden blir slik:

```
if(alleVerdsdelane[tilfeldigTal].equals(verdsdel)){
    text("RIKTIG", 950, 50);
}
```

Sjekkliste

- [ ] Skriv inn `if` -setninga som me akkurat forklarte.

- [ ] Sjekk at koden fungerer. No skal teksten "RIKTIG" visast på skjermen når du trykkar på riktig verdsdel.

Her er koden så langt.

```
PImage verdskartet;
String[] alleVerdsdelane = {"ASIA", "EUROPA", "NORD-AMERIKA",
  "SØR-AMERIKA", "OSEANIA", "AFRIKA"};
int tilfeldigTal;

void setup(){
  size(1200, 850);
  verdskartet = loadImage("world-map.png");
  tilfeldigTal = (int)random(6);
}
void draw(){
  image(verdskartet, 0, 0);
  text("Trykk på: " + alleVerdsdelane[tilfeldigTal], 50, 50);

  sjekkVerdsdel("EUROPA", 365, 694, 455, 33);
  sjekkVerdsdel("ASIA", 694, 1197, 537, 88);
  sjekkVerdsdel("AFRIKA", 493, 695, 708, 455);
  sjekkVerdsdel("AFRIKA", 694, 740, 698, 537);
  sjekkVerdsdel("OSEANIA", 874, 1194, 780, 537);
  sjekkVerdsdel("NORD-AMERIKA", 2, 365, 528, 36);
  sjekkVerdsdel("SØR-AMERIKA", 238, 449, 810, 528);
}

void sjekkVerdsdel(String verdsdel, int hogreGrense, int venstreGrense,
  int nedreGrense, int ovreGrense){
    if(mouseX > hogreGrense && mouseX < venstreGrense &&
      mouseY < nedreGrense && mouseY > ovreGrense)]{
        if(alleVerdsdelane[tilfeldigTal].equals(verdsdel)){
          text("RIKTIG", 950, 100);
        }
    }
}
```

Steg 9: Få ny verdsdel å trykke på

Det er jo litt kjipt å måtte starte programmet på nytt kvar gong spelaren har trykka på riktig verdsdel. Difor må me velje eit nytt tilfeldig tal i `if` -setninga som sjekkar om spelaren trykka på riktig verdsdel.

Då gjer me akkurat det same som me gjorde i `setup` -metoden då me sette `tilfeldigTal` til å vere eit tilfeldig tal.

Sjekkliste

- [ ] Set `tilfeldigTal` til å bli eit nytt tilfeldig tal i `if` -setninga som sjekkar om spelaren har trykka på riktig verdsdel.

- [ ] Køyr koden og sjå at det fungerer. No skal du få opp ein ny verdsdel kvar gong du trykkar riktig.

Her er koden så langt. Sidan me berre har lagt til ei kodelinje inni metoden `sjekkVerdsdel` viser me berre den metoden:

```
void sjekkVerdsdel(String verdsdel, int hogreGrense, int venstreGrense,
  int nedreGrense, int ovreGrense){
    if(mouseX > hogreGrense && mouseX < venstreGrense &&
      mouseY < nedreGrense && mouseY > ovreGrense)]{
        if(alleVerdsdelane[tilfeldigTal].equals(verdsdel)){
          text("RIKTIG", 950, 100);
          tilfeldigTal = (int)random(6);
        }
    }
}
```

Steg 10: Få poeng

Det siste me skal gjere er å gi spelaren poeng kvar gong riktig verdsdel blir trykka på. Sidan me heile tida vil få fleire og fleire poeng, så må me lage ein variabel som me kan oppdatere når spelaren trykkar på riktig verdsdel.

Variabelen skal vere av typen `int` og kan til dømes heite `poeng` .

Sjekkliste

- [ ] Deklarer ein variabel av typen `int` med namnet `poeng` saman med dei andre deklareringane du har skrive i toppen av koden. Hugs at du kan gi variabelen eit anna namn viss du vil.

- [ ] Set variabelen til å vere `0`. Det gjer du ved å skrive denne kodelinja i `setup`-metoden:

```
    poeng = 0;
```

- [ ] Få poenga til å visast i vindauget ved å gjere eit metodekall på `text();`. Den fyrste parameteren må vere ein passande tekst i hermeteikn, og så `+ poeng`. Dei to neste bestemmer du sjølv.

- [ ] Test at koden fungerer så langt. No skal du sjå poenga på skjermen, men førebels får du ikkje fleire poeng sjølv om du trykkar på riktig verdsdel.

- [ ] Prøv å skrive den neste kodelinja der du trur den skal vere. Den gjer at kvar gong programmet les den, så blir verdien til `poeng` ein større. Her er kodelinja:

```
    poeng = poeng + 1;
```

- [ ] Test om koden fungerer. Viss du plasserte kodelinja riktig, så skal du få poeng kvar gong du trykkar på riktig verdsdel.

Nå gjenstår det berre å gi minuspoeng viss ein trykkar på feil verdsdel. Munnleg blir koden slik:

```
Viss verdsdelen som er trekt tilfeldig er lik verdelsdelen frå Stringe
n,
    Så visar me teksten "RIKTIG" i vindauget.
    trekk eit nytt tilfeldig tal
    få eit poeng
elles
    så skal me få eit minuspoeng
```

Me har allereie dei `if`-setningane som utfører sjekken, og me har alt som skjer viss me trykkar på riktig verdsdel. Det einaste som manglar er kva som skjer frå `elles` og nedover.

Når me skriv ei `if`-setning kan me alltid leggje til ein `ellers` (`else` på engelsk). Viss me legg til ein `else` etter ei `if`-setning, så vil koden i `else` skje viss testen i `if`-setninga er usann. Så viss ein trykkar på riktig verdsdel får ein poeng, elles får ein minuspoeng.

Ein `else` blir åpna og lukka på same måte som ei `if`-setning, altså med krøllparentesar: `{` og `}`.

Sjekkliste

- [ ] Legg til ein `else` rett etter at `if`-setninga som sjekkar om verdsdelane er like er avslutta.

- [ ] Kodelinja som du brukte for å leggje til eit poeng kan brukast til å trekkje frå eit poeng. Alt du må endre er å skrive `-` i staden for `+`. Skriv denne kodelinja inni `else`.

- [ ] Sjekk at koden køyrer. Kva skjer? Får du eit minuspoeng kvar gong du trykkar på feil verdsdel? Kva skjer når du trykkar på riktig verdsdel, får du poeng?

Grunnen til at du får for mange minuspoeng er at du ikkje rekk å sleppe knappen på musepeikaren før programmet har endra kva verdsdel du skal trykke på. Programmet køyrer gjennom all koden din om lag 30-60 gonger i sekundet. Det tyder at du må trykke veldig raskt for å ikkje få minuspoeng kvar gong du trykkar, sjølv om du trykkar på riktig verdsdel. Du kan sjekke kor raskt koden blir køyrt ved å gjere eit metodekall på `text();` og så setje den fyrste parameteren til å vere `frameRate`. Talet du ser på skjermen er antal gonger koden blir køyrt kvart einaste sekund.

Me kan fikse alle minuspoenga ved å gjere eit metodekall på `delay();`. Denne gjer at koden tek ei lita pause. Parameteren er antal millisekund koden skal ta pause. Eitt sekund er det same som 1000 millisekund. Du må vere veldig forsiktig med å bruke `delay();` til vanleg, fordi den pausar *heile* programmet, så absolutt ingenting blir sjekka så lenge programmet har pause.

Sjekkliste

- [ ] Gjer eit metodekall på `delay();` rett etter at du har avslutta `else`. La den ta 50 millisekund som parameter.

- [ ] Test at koden køyrer.

- [ ] Bytt til 5000 millisekund og test programmet att. Kva skjer?

- [ ] Prøv med andre tider, og finn ut kva du synest er best.

- [ ] Viss du ikkje har gjort det endå, så må du bytte tilbake til biletet av verdskartet utan boksane som markerer grensene våre.

Det var det! No har du eit lite spel. Utvid spelet slik du vil sjølv. Det er mange moglegheiter. Under finn du nokre ekstraoppgåver du kan gjere viss du har lyst.

Her er koden så langt. Hugs at koden din kan vere litt annleis viss du har valt andre variabelnamn eller verdiar enn det me har. Det viktigaste er at den fungerer!

```
PImage verdskartet;
String[] alleVerdsdelane = {"ASIA", "EUROPA", "NORD-AMERIKA",
  "SØR-AMERIKA", "OSEANIA", "AFRIKA"};
int tilfeldigTal;
int poeng;

void setup(){
  size(1200, 850);
  verdskartet = loadImage("world-map.png");
  tilfeldigTal = (int)random(6);
  poeng = 0;
}
void draw(){
  image(verdskartet, 0, 0);
  text("Trykk på: " + alleVerdsdelane[tilfeldigTal], 50, 50);
  text("Poeng: " + poeng, 50, 100);

  sjekkVerdsdel("EUROPA", 365, 694, 455, 33);
  sjekkVerdsdel("ASIA", 694, 1197, 537, 88);
  sjekkVerdsdel("AFRIKA", 493, 695, 708, 455);
  sjekkVerdsdel("AFRIKA", 694, 740, 698, 537);
  sjekkVerdsdel("OSEANIA", 874, 1194, 780, 537);
  sjekkVerdsdel("NORD-AMERIKA", 2, 365, 528, 36);
  sjekkVerdsdel("SØR-AMERIKA", 238, 449, 810, 528);
}

void sjekkVerdsdel(String verdsdel, int hogreGrense, int venstreGrense,
  int nedreGrense, int ovreGrense){
    if(mouseX > hogreGrense && mouseX < venstreGrense &&
      mouseY < nedreGrense && mouseY > ovreGrense)]{
        if(alleVerdsdelane[tilfeldigTal].equals(verdsdel)){
          text("RIKTIG", 950, 100);
          tilfeldigTal = (int)random(6);
          poeng ++;
        } else {
          poeng --;
        }
        delay(80);
    }
}
```

Steg 11: Ekstra

Dette er ekstraoppgåver du kan gjere viss du vil.

Prøv dette

- [ ] Legg inn teksten som viser kva koordinatane for musepeikaren er att.

Forbetre verdskartet

- [ ] Finn koordinatane for tuppen av India og Sri Lanka, og skriv kode som gjer at dette området blir rekna med som Asia.
- [ ] Finn ut nøyaktig kva øyer mellom Oseania og Asia som høyrer til kva verdsdel, og fiks grensa mellom dei to verdsdelane slik at den blir meir riktig.
- [ ] Skriv kode som gjer at Nord-Amerika ikkje inkluderer den vestlege biten av Grønland, men framleis inkluderer alt av Nord-Amerika.
- [ ] Fiks koden slik at heile Grønland høyrer til Europa.

Fylker i Noreg

- [ ] Bytt ut verdskartet med eit kart over Noreg.
- [ ] Lag firkantar som du brukar til å skilje fylka våre.
- [ ] Skriv kode som sjekkar om musepeikaren er innanfor fylka i Noreg.

Heilt andre oppgåver

- [ ] Finn eit kart for Game of Thrones, og lag eit spel der spelaren kan gjette kor dei ulike folkeslaga høyrer heime.

- [ ] Finn eit bilete av ulike dyr, lag ei liste med namna på dyra på engelsk, eller eit anna språk. Lag grenser rundt dyra og la spelaren gjette kva dei ulike dyra heiter på ulike språk.

- [ ] Finn eit bilete av ein menneskekropp som er laga for undervisning, slik at du kan sjå det som er inni. Lag grenser rundt det du ser, til dømes hjartet, lungene, nyrene, ribbeina og liknande. La spelaren gjette kor dei ulike organa og beina er.

Lisens: CC BY-SA 4.0 (http://creativecommons.org/licenses/by-sa/4.0/deed)