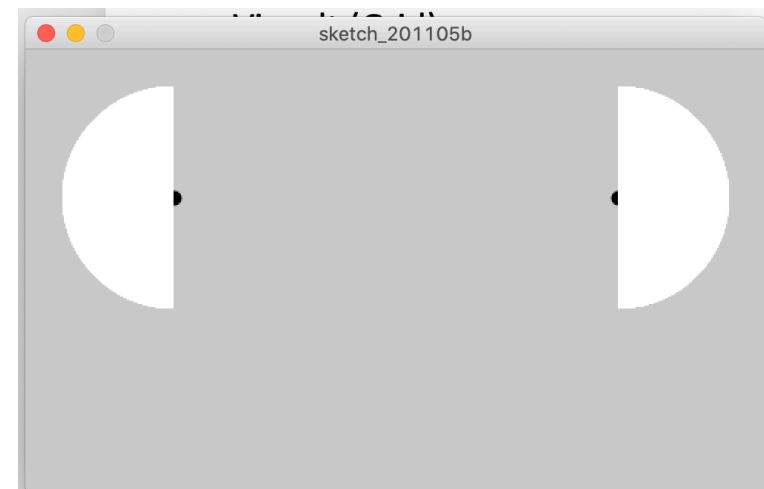
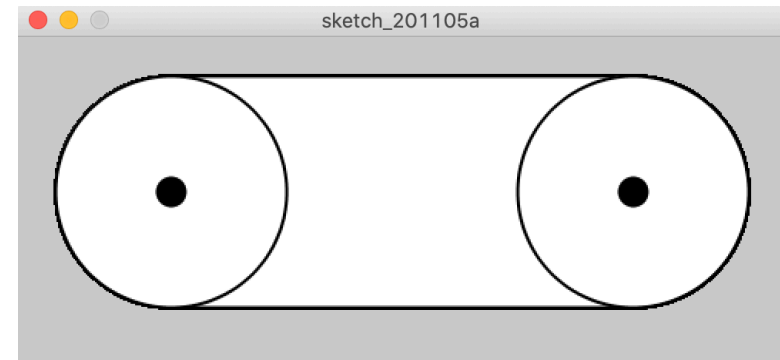
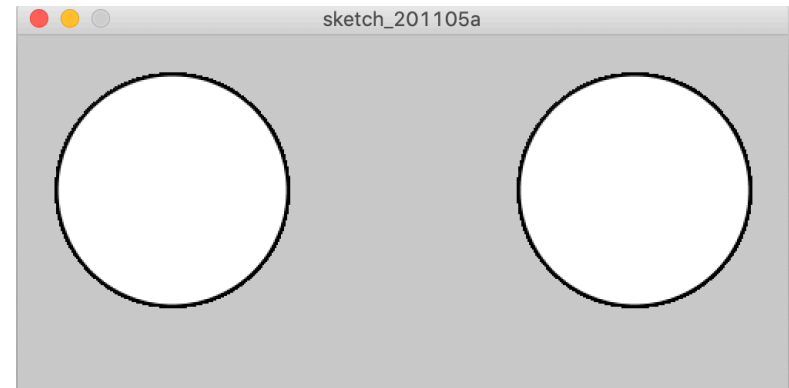
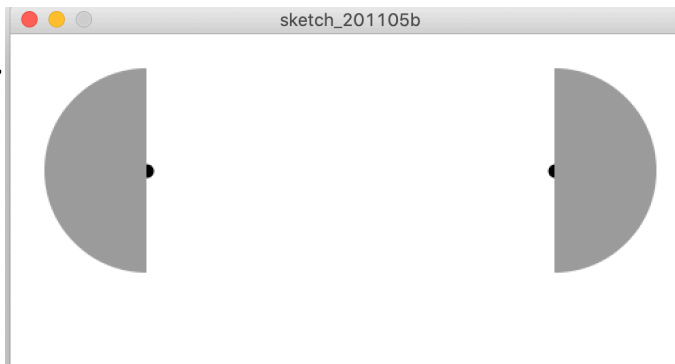


Processing med betingelser&loops

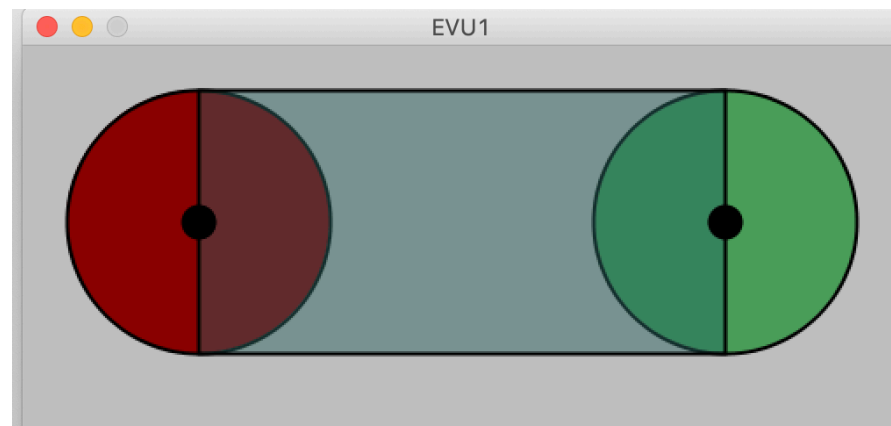
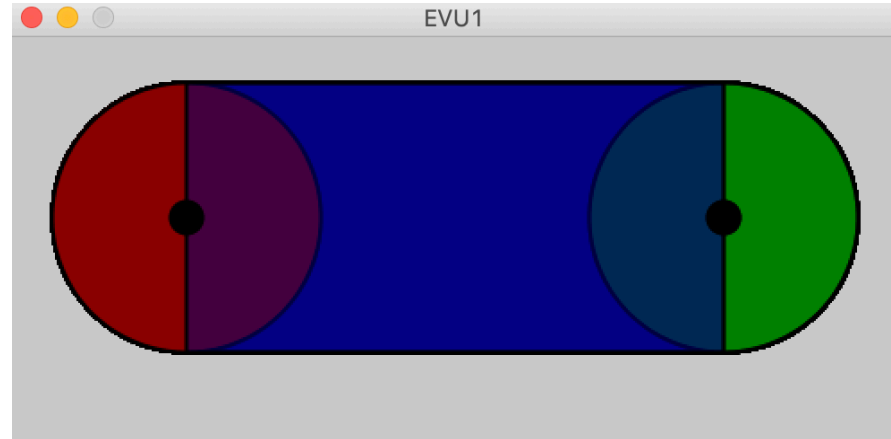
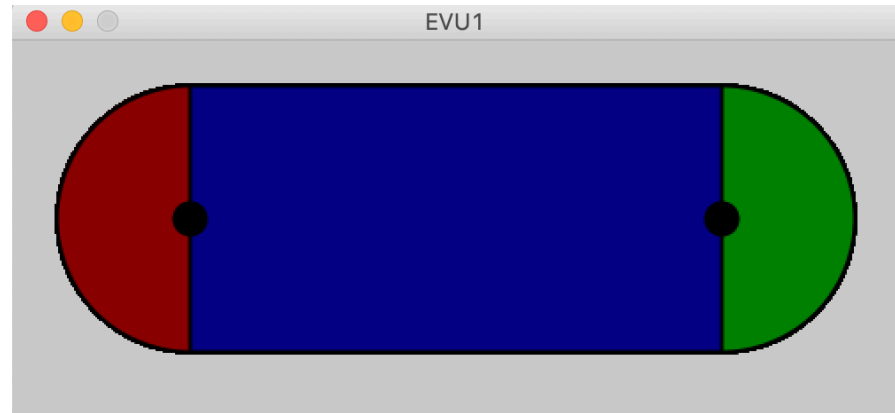
Pixels

- Visuelt (Grid)
- THE REFERENCE – (ø 2-8)
- Line(1,0,4,5)
 - Metode + argumenter
- Point, line, rectangle, ellipse
 - Location - koordinater
 - Size – width,height
 - Color – fill, stroke, strokeWeight
- Rect(2,3,w,h)
- Ellipse(1
- Arc



Color

- Grayscale (0-255)
- RGB (r,g,b)
- Transparency (r,g,b,0-255)
- HSB (h,s,b)
- HSB(0-360,0-100,0-100)



Processing - tooling

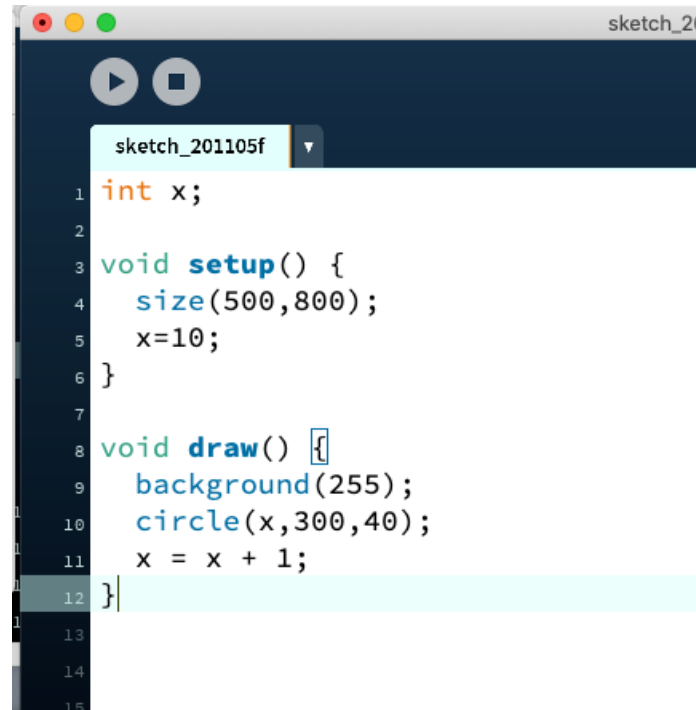
- preferences
- Sketch
- First sketch
 - **Function call**
 - Assignment
 - Control
- Functions
 - Build in
 - Size
 - Println
 - Comments //
 - Errors (ø 2-6)



Interaction

- The Flow

- Setup
 - Draw
 - Internal loop
 - (framecounter)
- Block of code
- Mouse
 - mouseX,mouseY
 - Ex 3-2 (background)
 - mouseX,mouseY
 - Ex 3-4

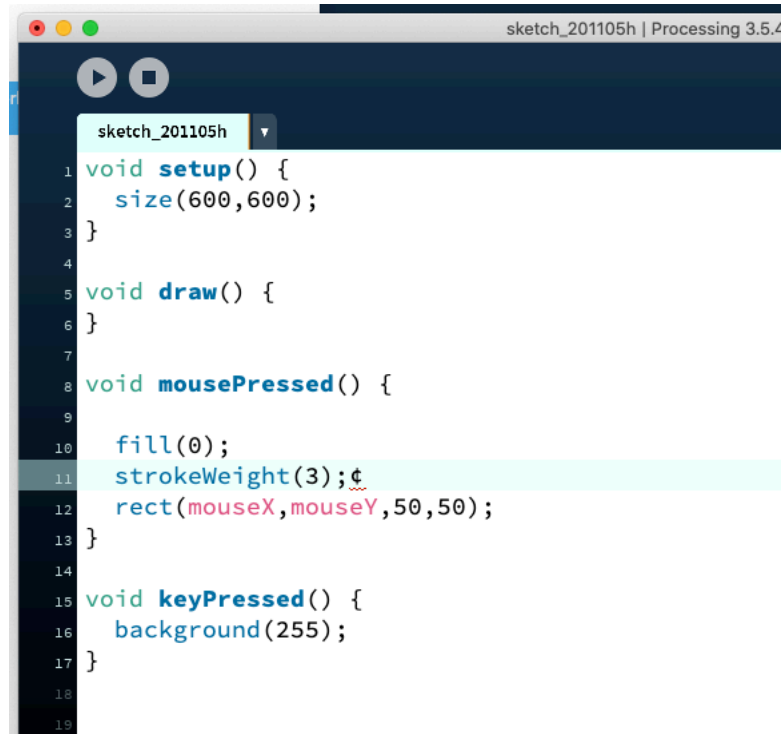


```
sketch_201105f
1 int x;
2
3 void setup() {
4   size(500,800);
5   x=10;
6 }
7
8 void draw() {
9   background(255);
10  circle(x,300,40);
11  x = x + 1;
12 }
```

Mere mus ..

- Interaction

- mousePressed()
- mouseReleased()
- keyPressed()



```
sketch_201105h | Processing 3.5.4

1 void setup() {
2   size(600,600);
3 }
4
5 void draw() {
6 }
7
8 void mousePressed() {
9
10  fill(0);
11  strokeWeight(3);
12  rect(mouseX,mouseY,50,50);
13 }
14
15 void keyPressed() {
16   background(255);
17 }
18
19
```

Variabler & operatorer

- Brukt i processing

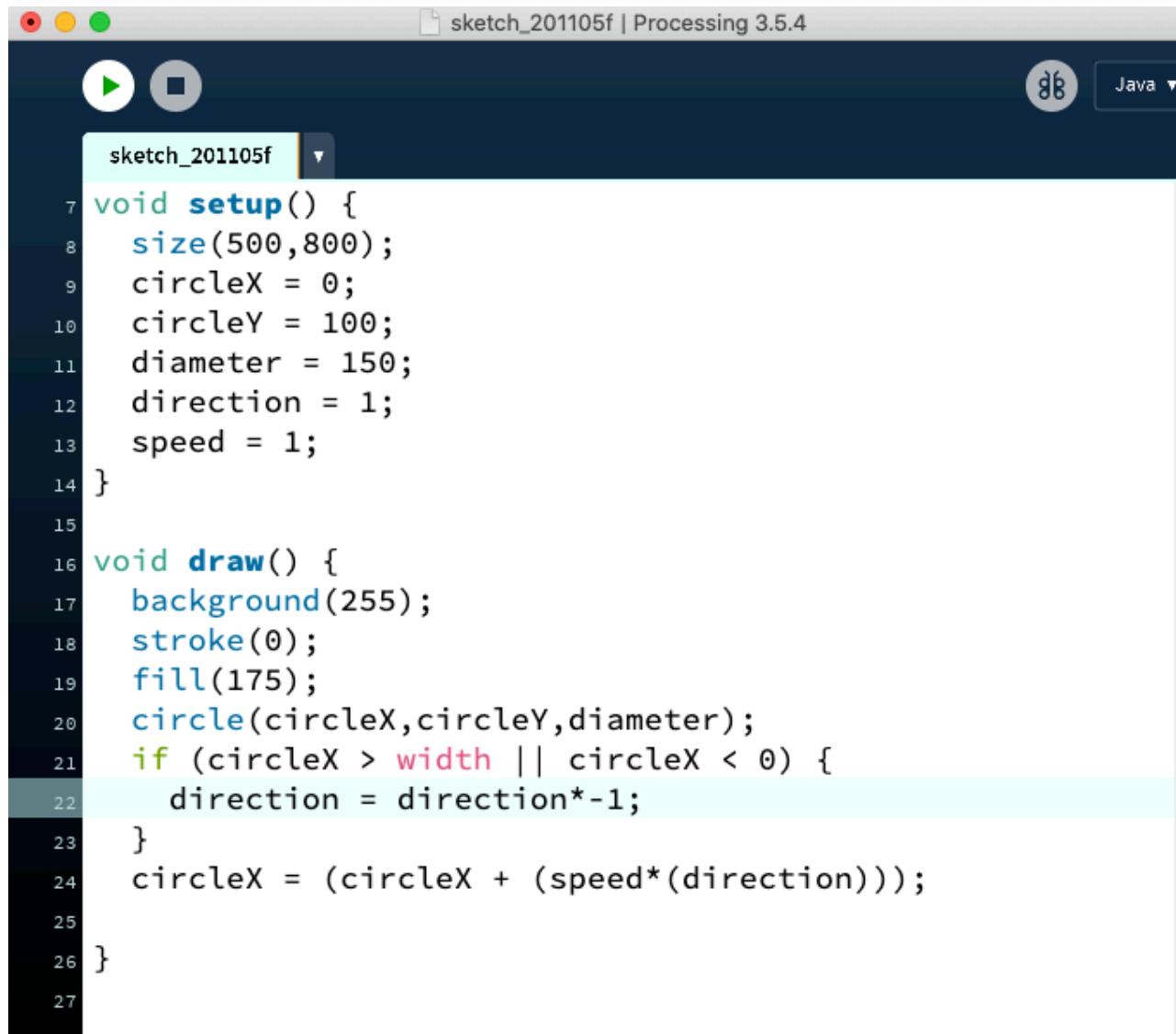
- boolean
- Int
- float
- char
- String

- System Variabler

- width —Width (in pixels) of sketch window.
- height —Height (in pixels) of sketch window.
- frameCount —Number of frames processed.
- frameRate —Rate that frames are processed (per second).
- key —Most recent key pressed on the keyboard.
- keyPressed —True or false? Is a key pressed?
- mousePressed —True or false? Is the mouse pressed?



```
sketch_201105f
1 int x;
2
3 void setup() {
4   size(500,800);
5   x=10;
6 }
7
8 void draw() {
9   background(255);
10  circle(x,300,40);
11  x = x + 1;
12 }
```

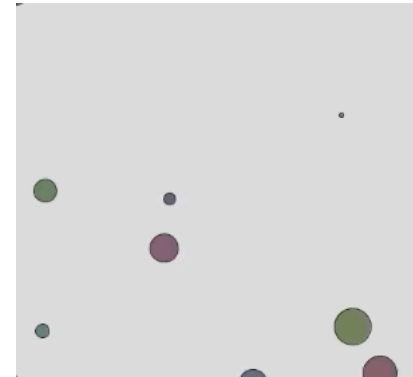


The image shows a screenshot of the Processing IDE window. The title bar at the top reads "sketch_201105f | Processing 3.5.4". Below the title bar is a dark blue toolbar with a green play button (run) and a grey square button (stop). On the right side of the toolbar is a circular icon with a stylized 'P' and a dropdown menu labeled "Java". Below the toolbar is a tab labeled "sketch_201105f". The main area is a code editor with a light blue background and a dark blue sidebar on the left showing line numbers from 7 to 27. The code is written in Java and defines two functions: `void setup()` and `void draw()`. The `setup()` function initializes variables: `size(500,800);`, `circleX = 0;`, `circleY = 100;`, `diameter = 150;`, `direction = 1;`, and `speed = 1;`. The `draw()` function contains `background(255);`, `stroke(0);`, `fill(175);`, and `circle(circleX, circleY, diameter);`. It also has an `if` statement: `if (circleX > width || circleX < 0) { direction = direction*-1; }`. Finally, it updates the circle's position: `circleX = (circleX + (speed*(direction)));`. The line `direction = direction*-1;` is highlighted with a light green background.

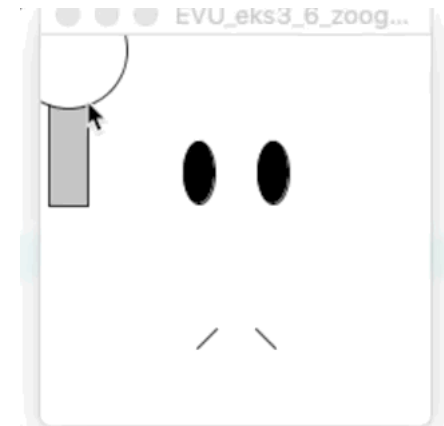
```
7 void setup() {  
8   size(500,800);  
9   circleX = 0;  
10  circleY = 100;  
11  diameter = 150;  
12  direction = 1;  
13  speed = 1;  
14 }  
15  
16 void draw() {  
17   background(255);  
18   stroke(0);  
19   fill(175);  
20   circle(circleX, circleY, diameter);  
21   if (circleX > width || circleX < 0) {  
22     direction = direction*-1;  
23   }  
24   circleX = (circleX + (speed*(direction)));  
25  
26 }  
27
```


Random – “a bit of fun”

```
sketch_201112e | Processing 3.5.4
1 boolean go;
2
3 void setup() {
4   size(600, 600);
5 }
6
7 void draw() {
8   colorMode(HSB);
9   fill(int(random(255)), 100, 100);
10  circle(int(random(width)), int(random(height)), int (random(50)));
11 }
```



- Action
 - random()
- Kombineres med objekter i arrays
- Øvelse: boolean go?



Kontrol: Conditionals & operators

- Brugt i processing
 - Boolean expressions (mouseX>10)
 - If, else-if og else
 - Logic (ø 5-5 s68)
 - If-else if (eks 5-3)
 - Loops
 - For og while
 - Draw()
- System Variabler

CONTROL

Relational Operators

== (equality)
> (greater than)
>= (greater than or
equal to)
!= (inequality)
< (less than)
<= (less than or equal
to)

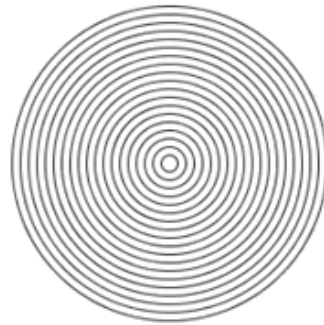
Logical Operators

&& (logical AND)
! (logical NOT)
|| (logical OR)

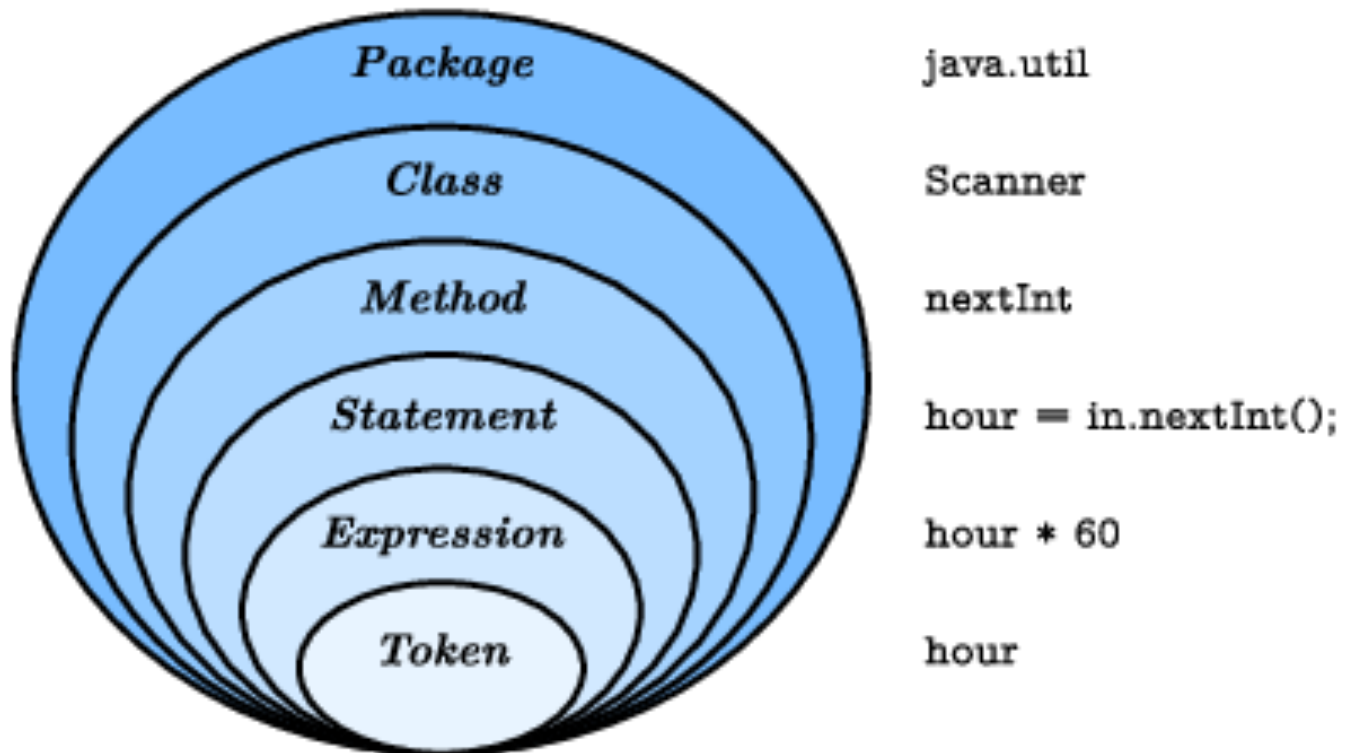
Conditionals

break
case
?: (conditional)
continue
default
else
if
switch()

Fælles øvelse – while and for



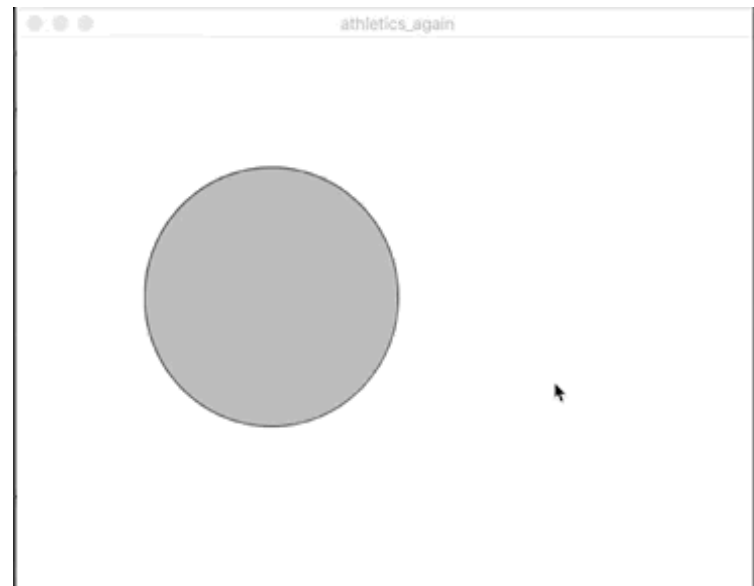
Elements of the language



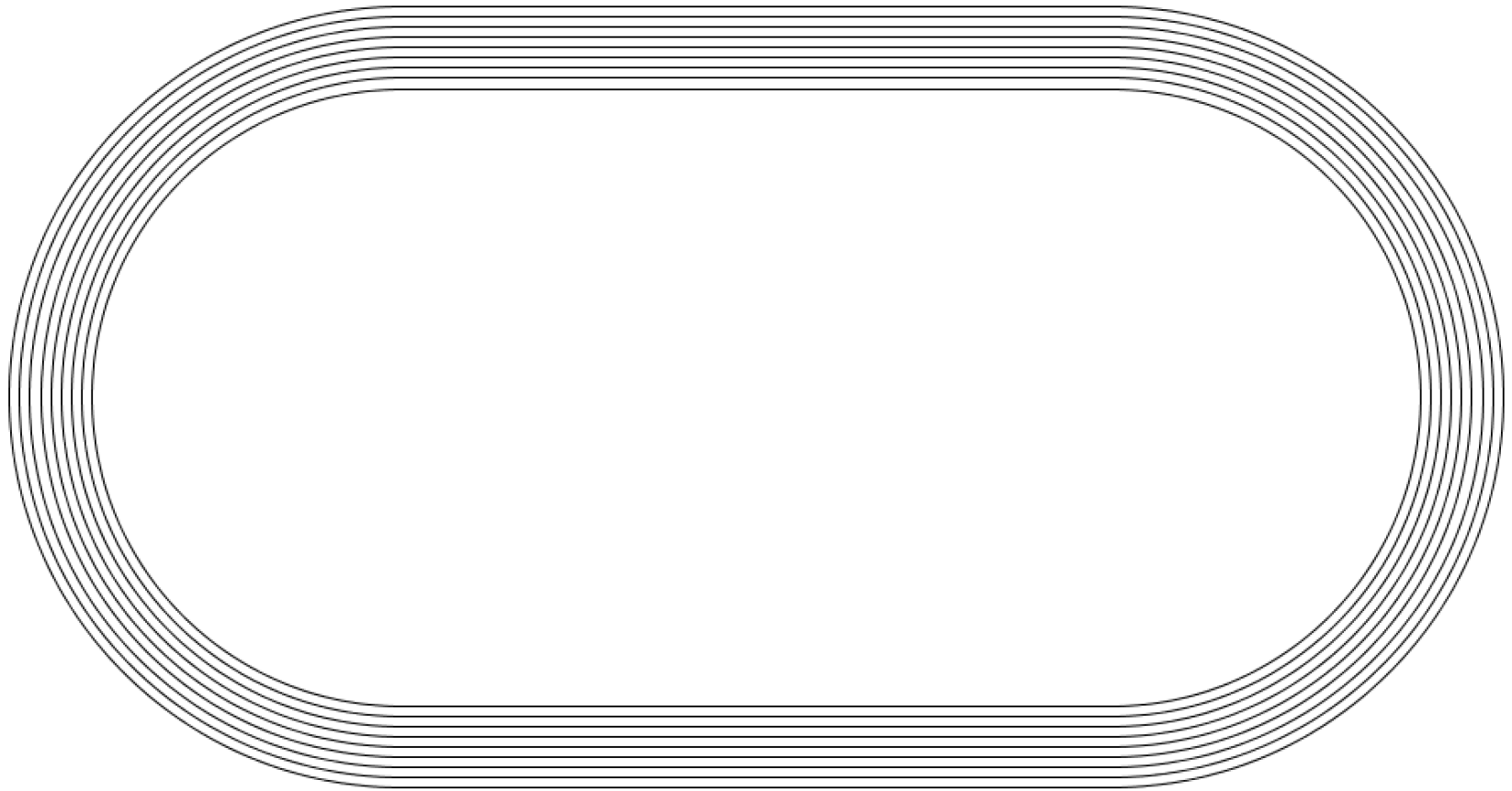
Operators og bogen s. 77

Level	Operators	Description	Associativity
15	() [] .	Function Call Array Subscript Member Selection	Left to Right
14	++ --	Postfix Increment / Decrement	Right to Left
13	++ -- + - ! ~ (type)	Prefix Increment / Decrement Unary plus / minus Logical negation / bitwise complement Casting	Right to Left
12	* / %	Multiplication Division Modulo	Left to Right
11	+ -	Addition / Subtraction	Left to Right
10	<< >> >>>	Bitwise Left Shift Bitwise Right Shift with sign extension Bitwise Right Shift with zero extension	Left to Right
9	< <= > >= instance of	Relational Less Than / Less than Equal To Relational Greater / Greater than Equal To Type Comparison for objects	Left to Right
8	== !=	Equality Inequality	Left to Right
7	&	Bitwise AND	Left to Right
6	^	Bitwise XOR	Left to Right
5		Bitwise OR	Left to Right
4	&&	Logical AND	Left to Right
3		Logical OR	Left to Right
2	?:	Conditional Operator	Right to Left
1	= += -= *= /= %= &= ^= = <<= >>=	Assignment Operators	Right to Left

øvelse - 5.5 side 77



Øvelse – Atletikbanen – step 2



7. Øvelse – Lav jeres egen pixel-quiz

- Find billeder indenfor et tema
- Læg dem i data-mappen
- Brug frameRate eller Counter til at udregne point

```
58  
59 void draw() {  
60     println("C " + factor);  
61     println("FrameRate: " + frameRate);  
    ..  
    ..  
    ..  
}
```