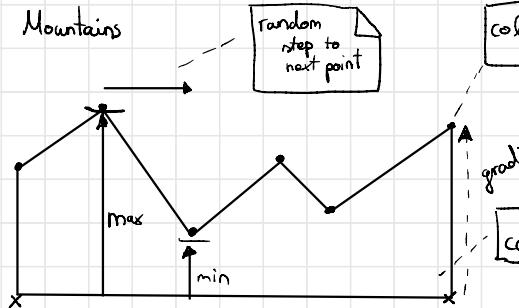
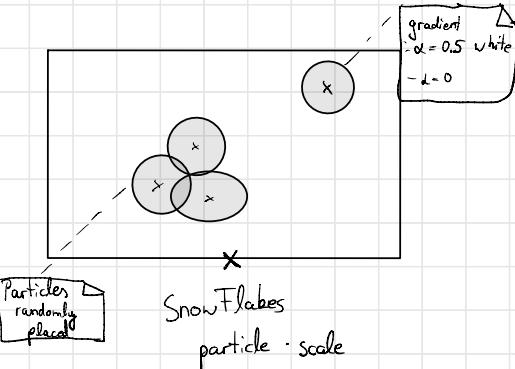
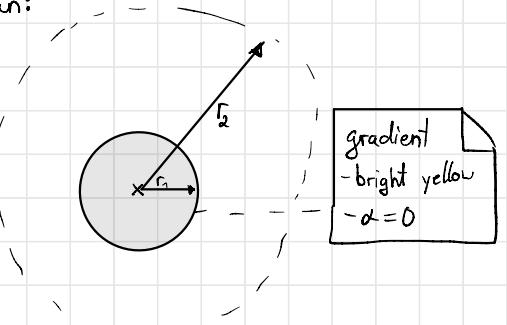


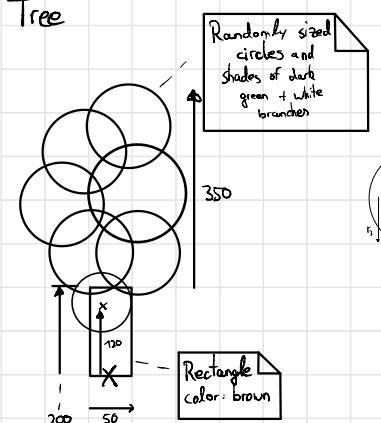
Mountains



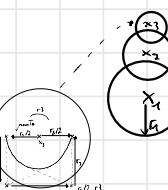
Sun:



Tree

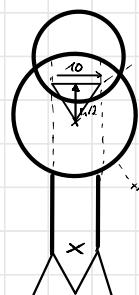
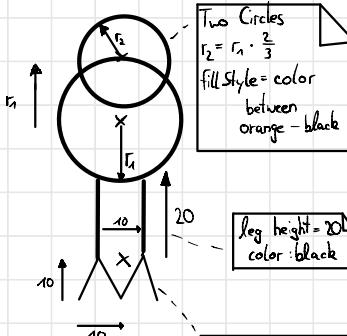


Snowman



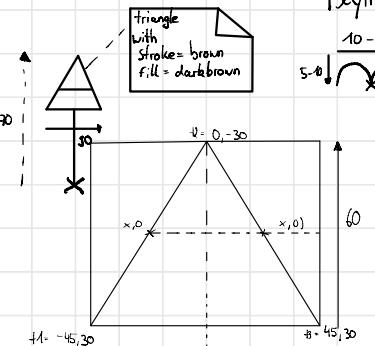
three circles getting smaller
translate $y = r_1$
 r_1 = random Num between 20 - 70
 $r_2 = r_1 / 3$
 $r_3 = r_2 / 3$

Sitting Bird

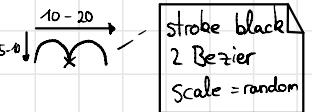


triangle
color: orange
height = $r_1 / 2$
width = 10
MoveTo: Center of x_1

Aviary

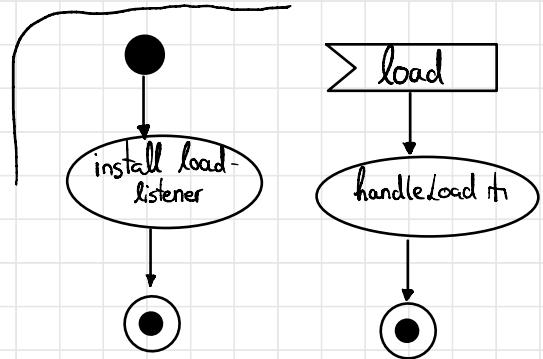


Flying Bird



MoveTo (-10, 10)
LineTo (-5, 0)
LineTo (0, 10) /color:
LineTo (5, 0) /black
LineTo (10, 10)
MoveTo (0, 10)

Activity-Diagramm: Aviary



Vector
x: number
y: number

handleLoad

get Rendering Context

draw Background

draw Sun (position)

draw Cloud (position, size)

2x draw Mountains
with different parameters

draw Trees

draw Snowman

draw Snowflakes (number, position)

draw Birds (n)

draw Aviary (position)

- position: Vector

r1: number = 30
r2: number = 150
gradient = Radial Gradient

Set color stops for gradient
transparent at 1.0
bright yellow at r1 / r2

Save transform

translate to position

draw full circle with r2

restore transform

draw Cloud

- position: Vektor
- size: Vektor
- nParticles: number
- radiusParticles: number

particle = Path with full circle
with radiusParticle
gradient: Radial with d=0.5
 $\rightarrow d=0$

Save transform

translate to -position



restore transform

[drawn < nParticles]

$x: \text{number} = (\text{random} - 0.5) \cdot \text{size.x}$
 $y: \text{number} = \text{random} \cdot \text{size.y}$

Save transform

translate x,y

restore transformation

draw Particle

draw SnowFlakes

- nFlakes: number
- position: vektor

Save Transform

translate to -position



[draw < nFlakes]

let x: number = random in
Flake space
let y: number = random in
Flake Space

draw Cloud({x, y}, {5, 5}, 1, 10)
+/-

draw Mountains

- position: Vektor
- min: number
- max: number
- colorLow: string
- colorHigh: string

StepMin: number = 10
StepMax: number = 50
x: number = 0

Save transform

translate to -position

moveTo(0,0)

lineTo 0, -max

X += random between
stepMin & stepMax

y: number = -min - Math.random(max - min)

lineTo(x, y)



restore Transform

draw Path

create gradient
with given Colors

line to x, 0

closePath

[x < canvas.width]

