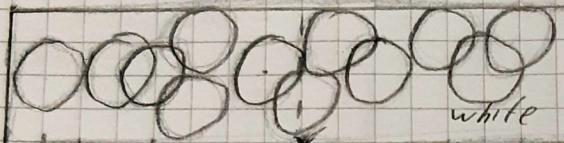


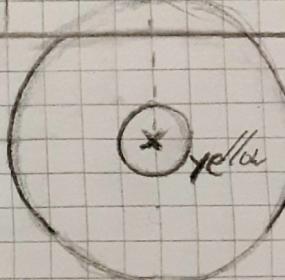
Wolke
 $(300/725)$
 $w = 400 \quad d = 700$

100 2 3 4 5



Sonne
 $(600/75)$
 $r_1 = 25 \quad r_2 = 100$

blau



Berge

Max H = 200/white

Max H = 150/white

Min H = 100/green

Min H = 75/grey

1

2

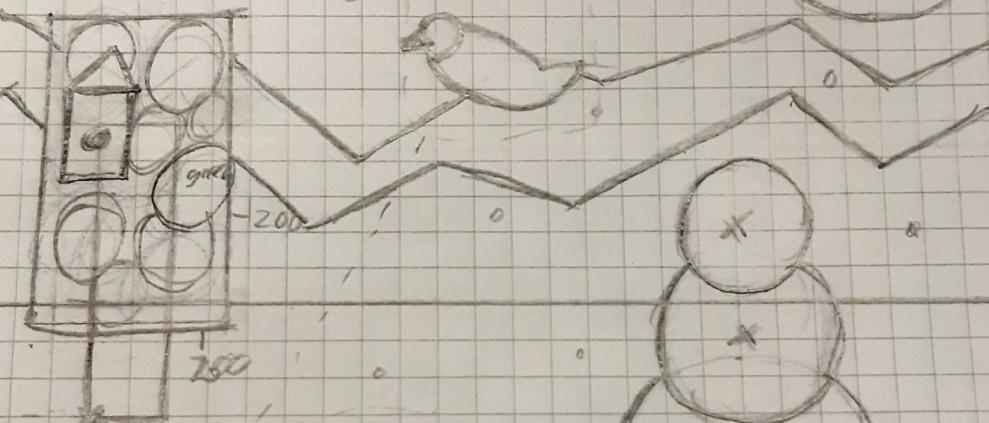
3

4

5

6

7

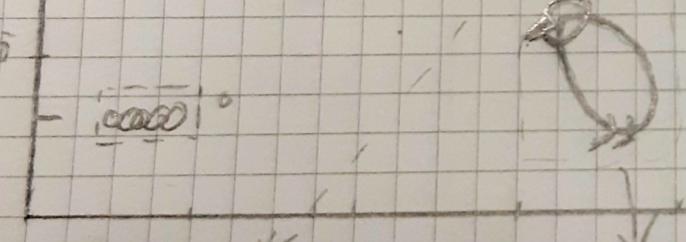


grüner
Schnitt

HSL (100, 20%, 90%)

Baum

(175/450, 50, -200)

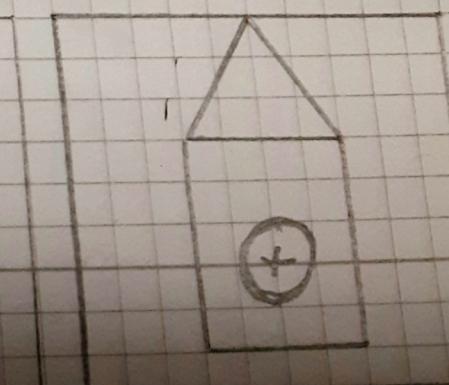
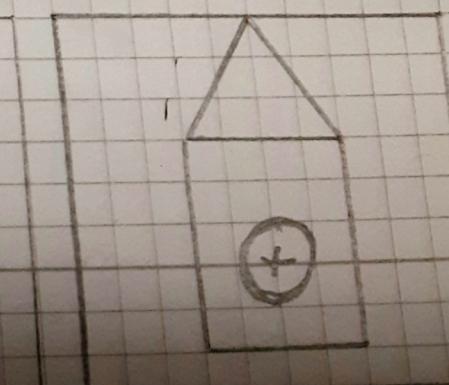


Food / brown

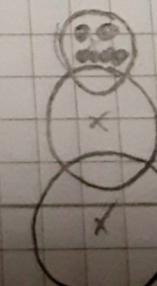
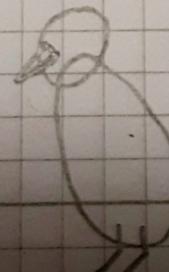
--

oooo

Schneeball



elipse
+ Kreis
+ Halbkreis
+ Dreieck



Canvas Rendering Context

Bird

color: string
aim: Vector
is Hungry: boolean
is Eating: boolean
is Hit: boolean

Constructor()
getRandomColor(): string
getFood(- mousePosition): Vector
eatFood(): void
changeDirection(): void
hitBird(- mousePosition: Vector)
draw(): void

Food

Constructor(- position: Vector)
draw(): void

Snowflake

Constructor()
draw(): void

Moveables

position: Vector
velocity: Vector

Constructor()

move(): void
draw(): void

Vector

x: number
y: number

Constructor(-x: number, -y: number): void

set(-x: number, -y: number): void

scale(-factor: number): void

add(-addend: Vector): void

random(-minLength: number, -maxLength: number): void

Snowball

aim: Vector

Constructor

targetBird(- mousePosition: Vector): void

reachedTarget(): void
draw(): void

~~load~~

```
let crc2 = canvasRenderingContext2D
let goldRatio = number = 0,62
let moveables = Moveable []
let highscore = number = 0
let url = string = "heroku server"
```

handleLoad ~~it~~

handleLoad

```
let canvas: HTMLCanvasElement = querySelector("canvas")
crc = canvas.getContext("2d")
```

draw everything static

```
let background = ImageData = crc2.getImageData(0, 0, 600)
```

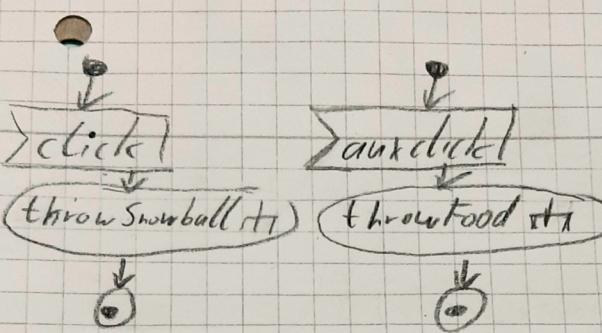
push X birds in moveables[] & draw them

push X Snowflakes in moveables[] & draw them

push Snowball in moveables[] & draw it

canvas.addEventListener("click", throwSnowball) ~~click~~

canvas.addEventListener("mousedown", throwFood) ~~mousedown~~



Interval
20s

Timeout
20s

X~~!~~
Update it

X~~!~~
end The Game

Update

-background: ImageData

crc.putImageData(-background, 0, 0)

draw & move all moveables[]

is a Bird && isHungry? → eat Food

is the Snowball? → reached Target

is a Bird && isHit? → Delete Bird

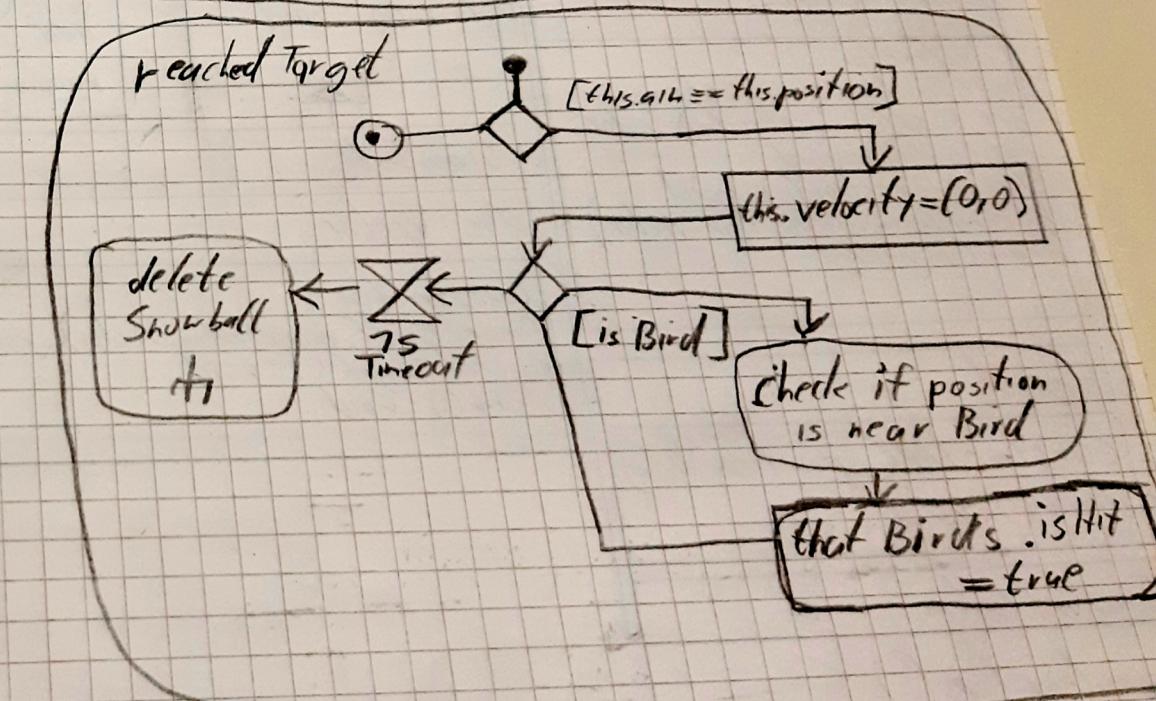
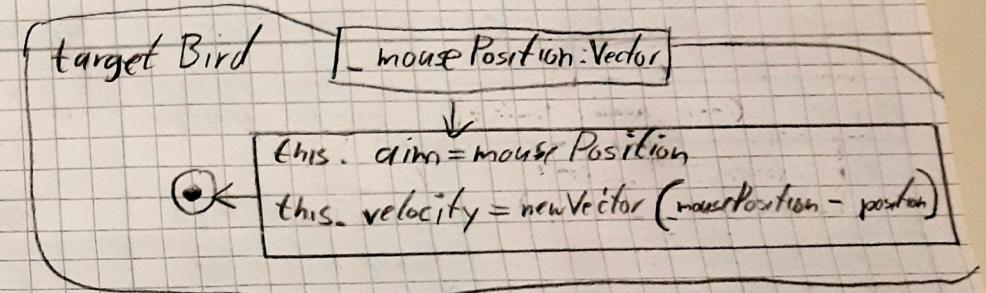
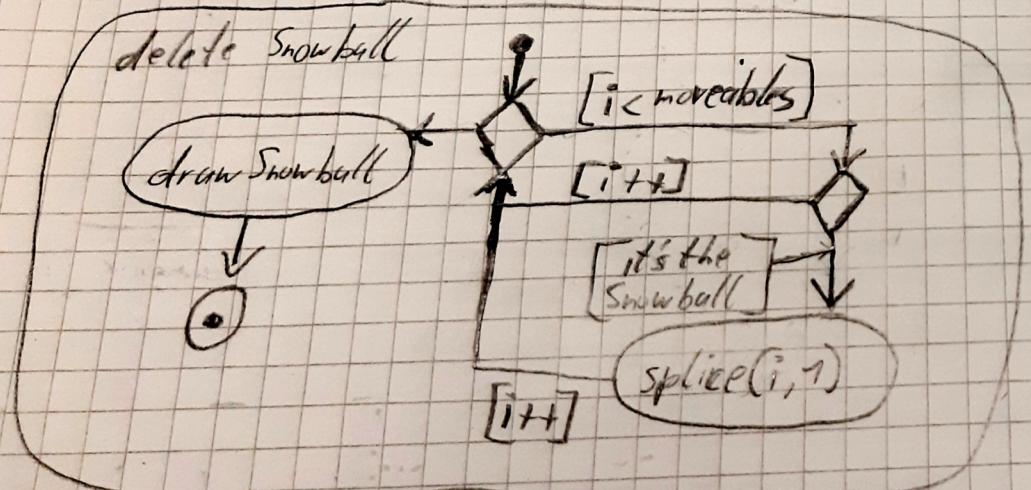
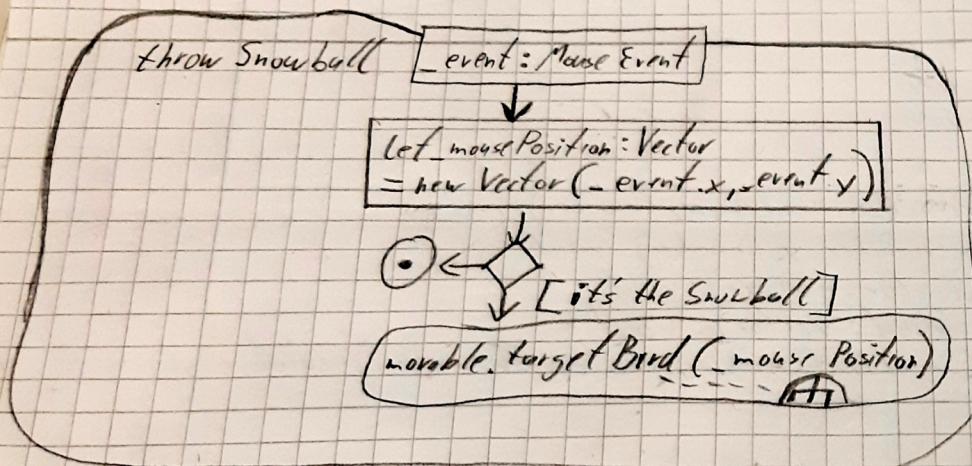
draw Score

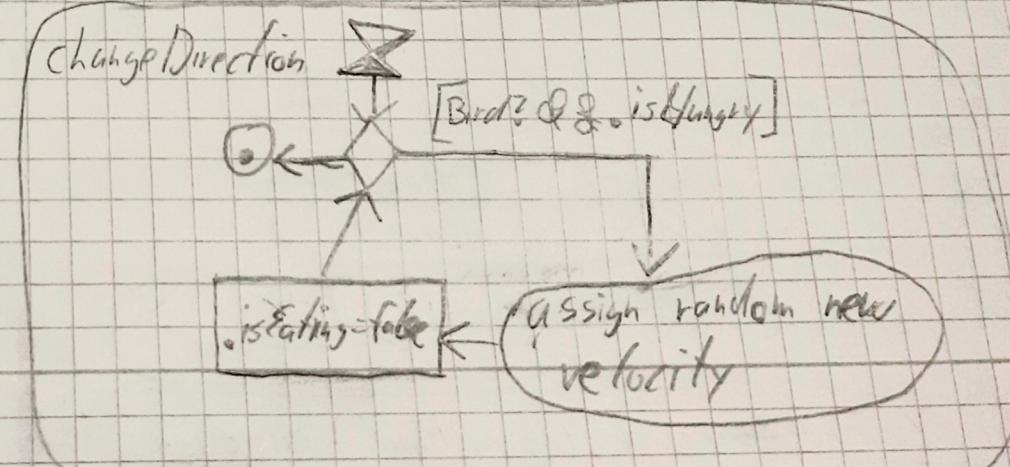
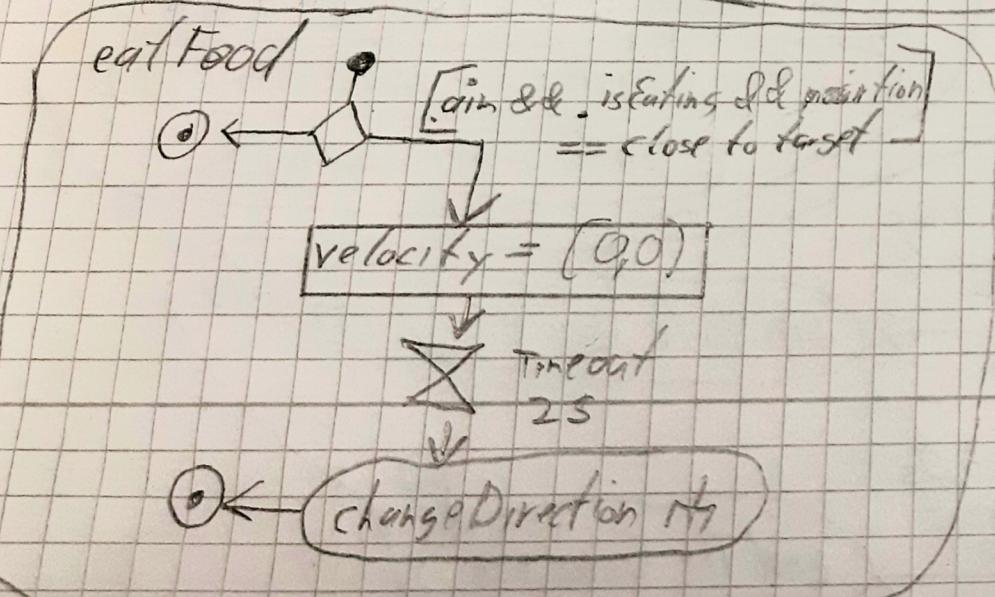
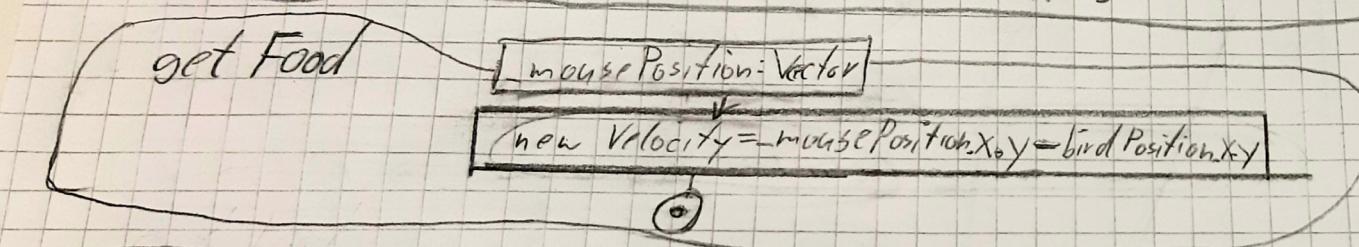
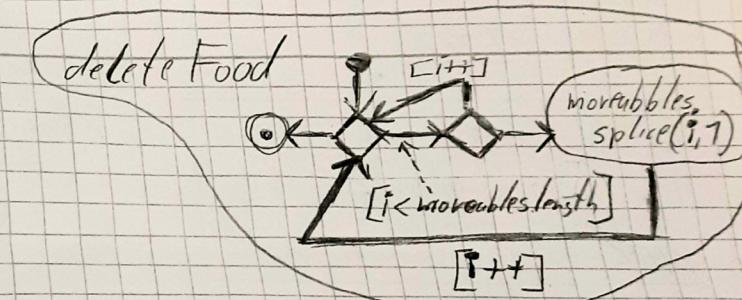
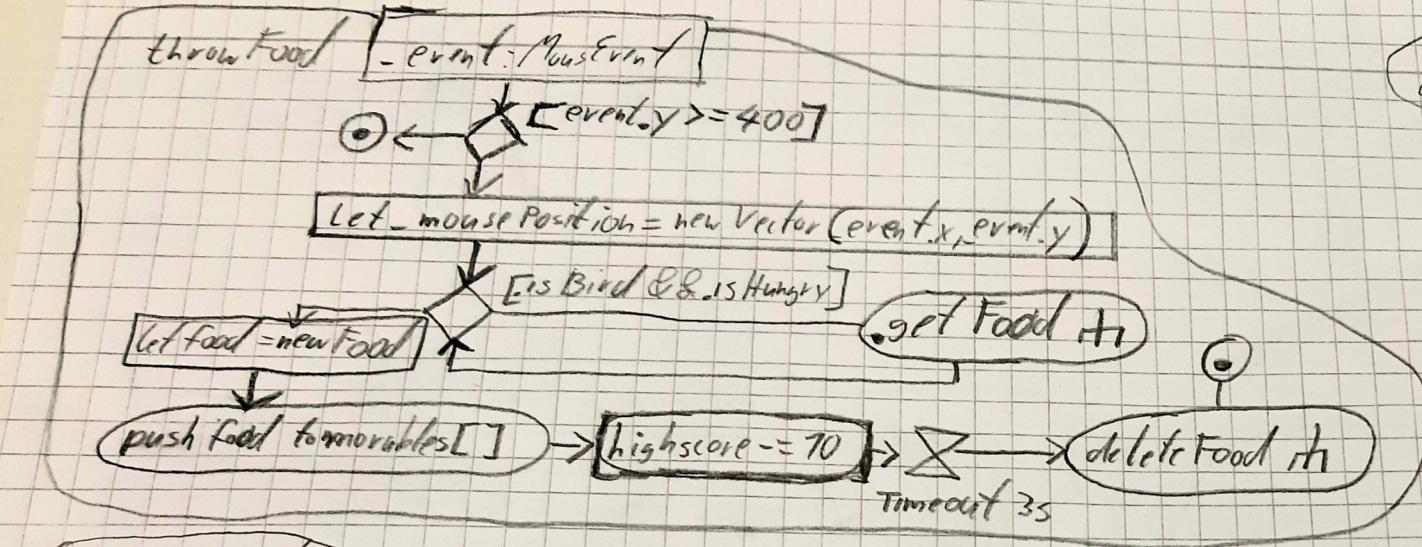
Alle 20ms
Interval

X → update [background], it

Nach 20s
Timeout

X → end The Game it





end The Game

let name : string = prompt("Enter your Name")

let query : string = "highscore=" + highscore + "&name=" + name

await fetch(url + "?" + query)

window.open("Link zur Start/Endseite")

Wird durch Button auf Startseite.html gestartet

display Highscore

let query : string = "command=retriev";

let response = await fetch(url + "?" + query)

let responseText : string = await response.text()

let highscoreList : HTMLDivElement = querySelector("div<server Response");

highscore.innerHTML = responseText