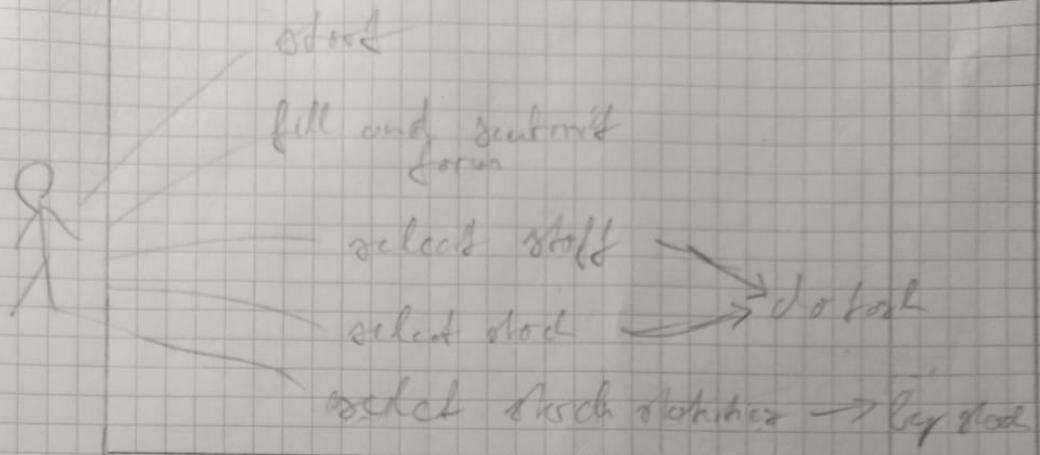




# Use Case



## Moveable

- + actived
- + position
- + direction
- + magnitude
- + properties

constructor  
(- position)

## Customer

# happiness

update happiness()

## Order

- + id : number
- + type : string
- + special wish : string
- + should deliver : bool
- + correct : number
- + customer : Customer
- + index : number

## Employee

- + assigned stock : Stock
- + starting task time : number
- + current task time : number
- + inventory : Stock[]
- + new stock : Stock[]
- + start Task()
- + check Time()
- + do Task()
- + display Inventory()
- + finish Task()

## Stock

- + name : string
- + amount : number
- + icon : string
- + price : number
- + position : Vector

constructor (- - - -)

## Class 2

Stod

name: string

color: color

icon: string

face: boolean

vector: Vector

constructor (name,

-color, -icon, -face,

-vector?)

select()

close (any?)

increase()

decrease()

draw()

Vector

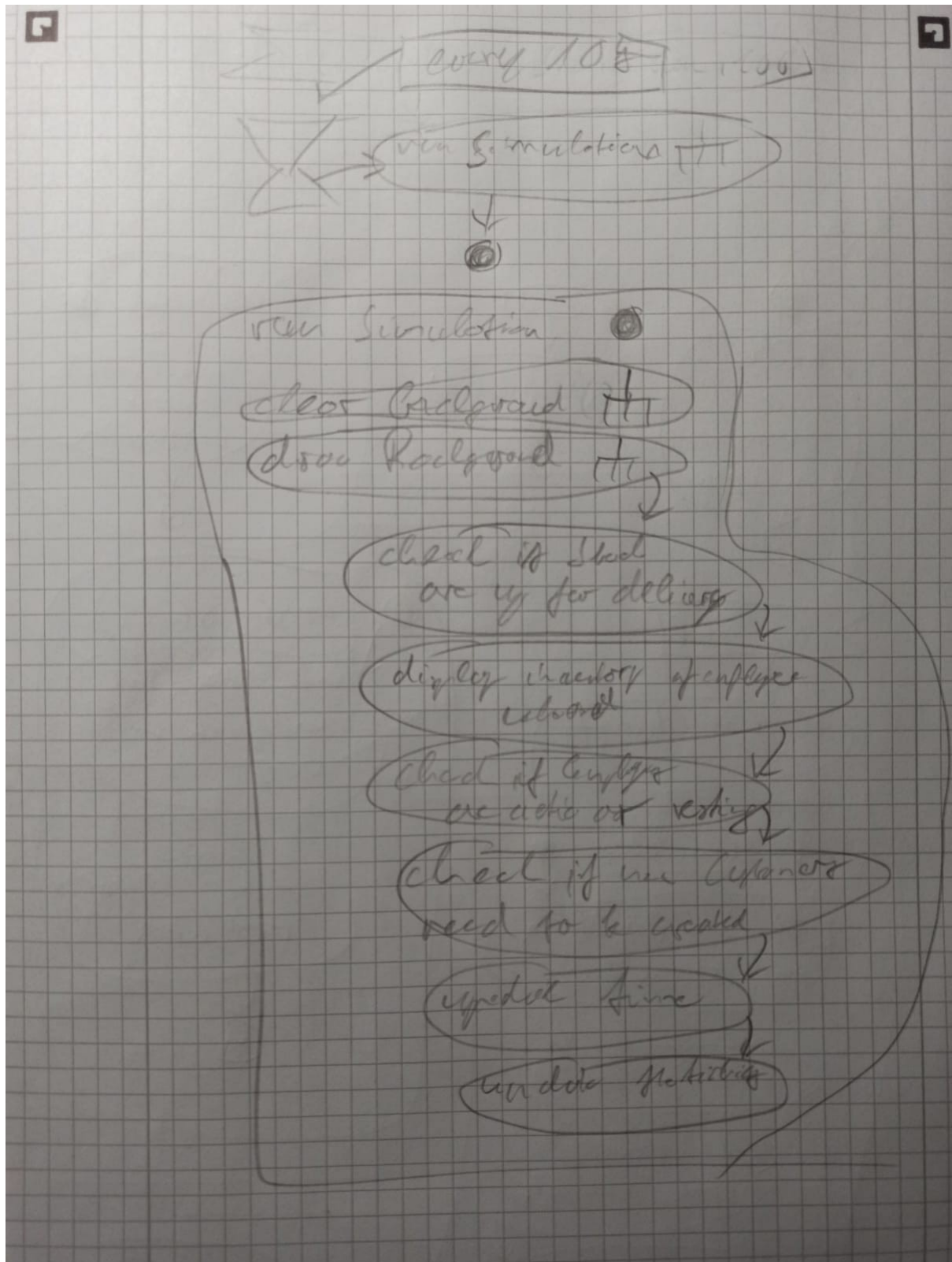
x: color

y: color

constructor

(-x: color, -y: color)

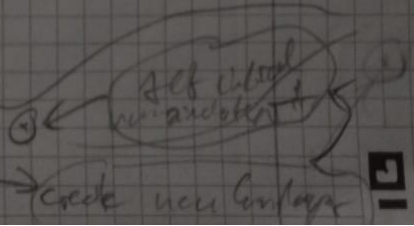
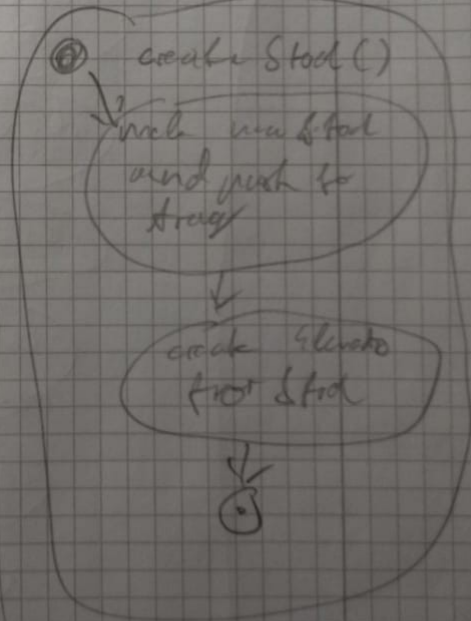
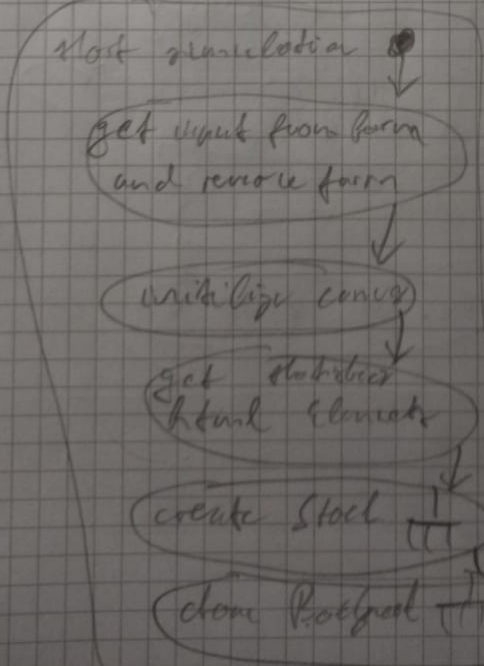
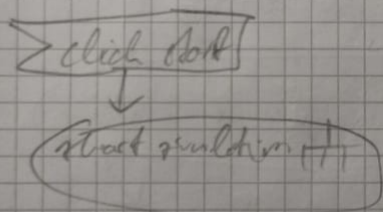
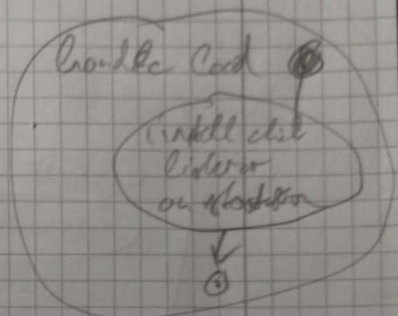
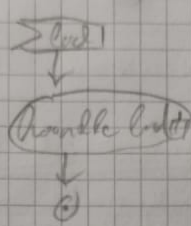
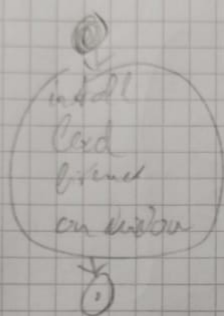
## Activity Diagram





# Abstract Diagram n. 1

Menu





Use Case

start

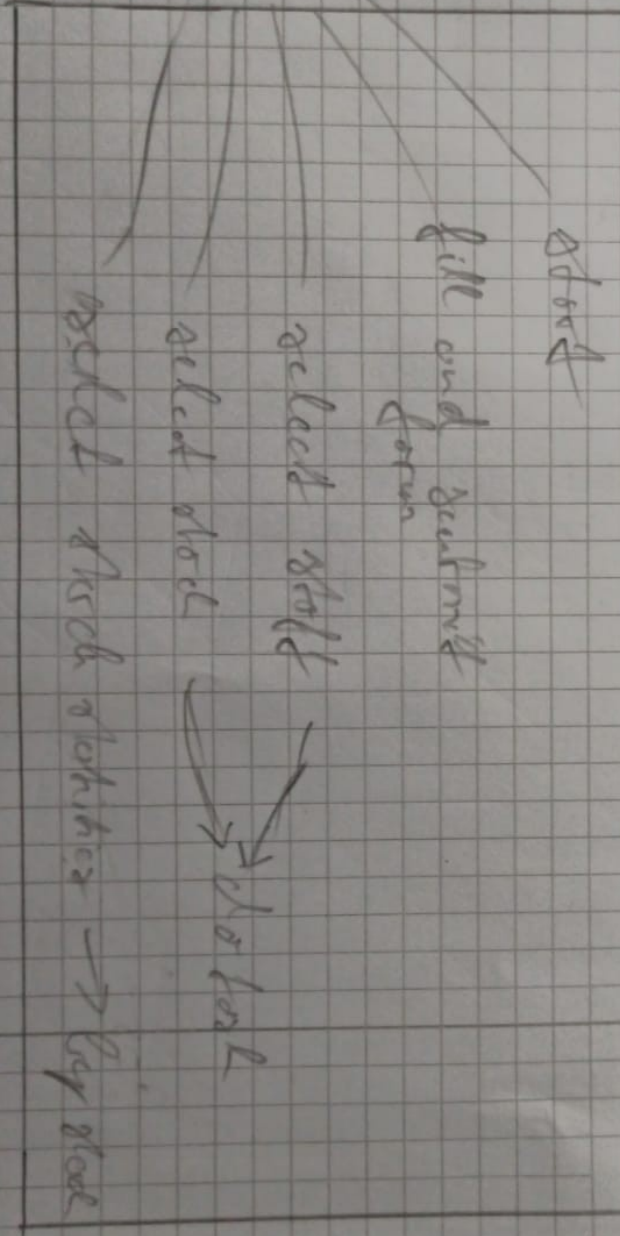
fill and generate form

select staff

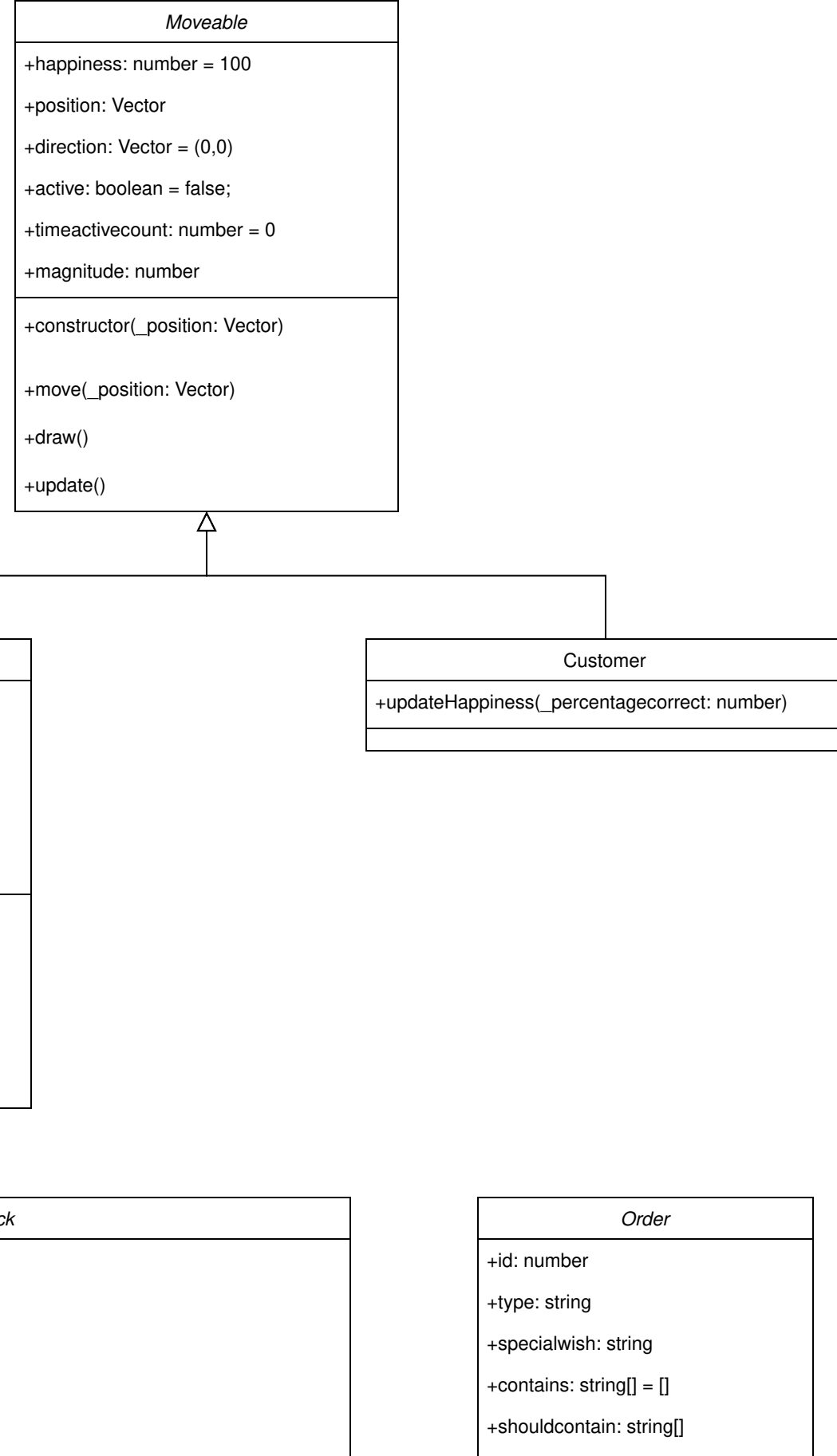
select stock

select stock statistics → by stock

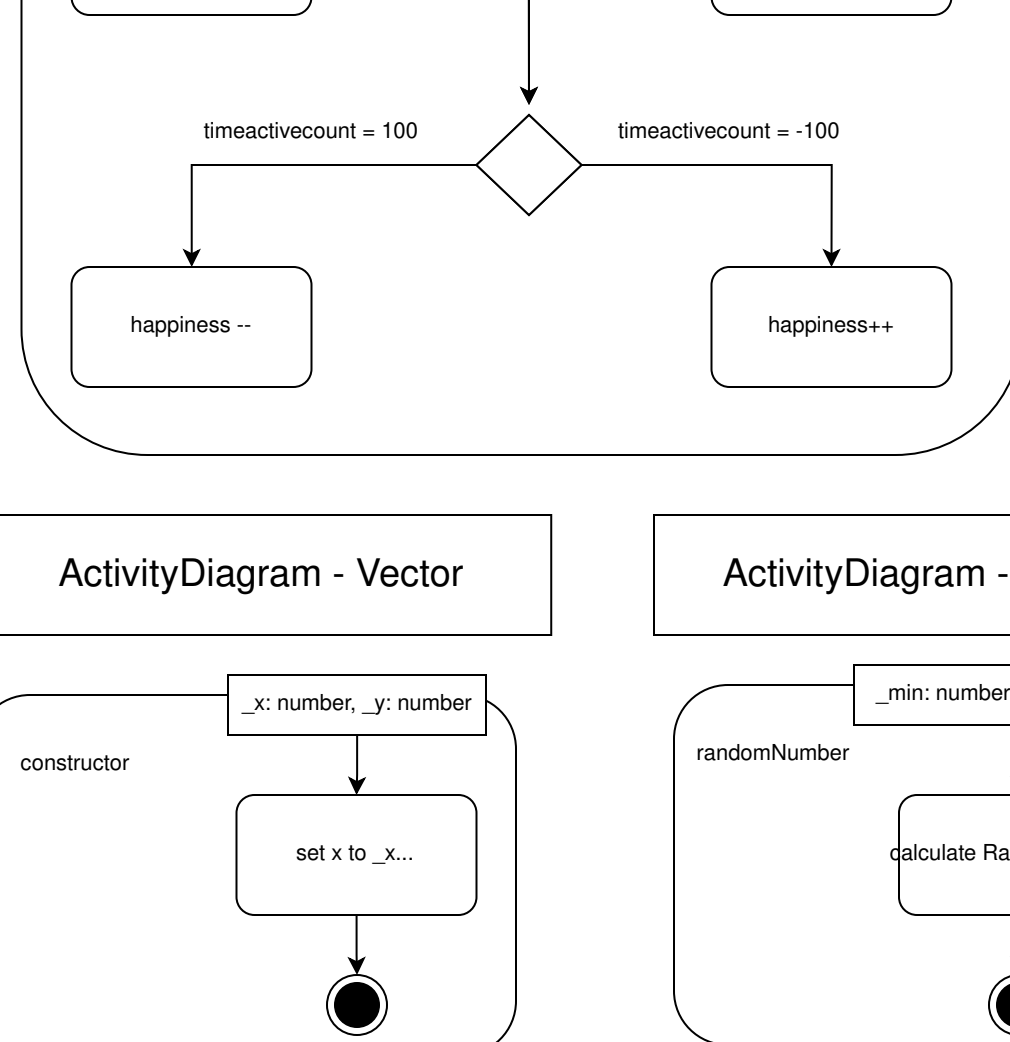
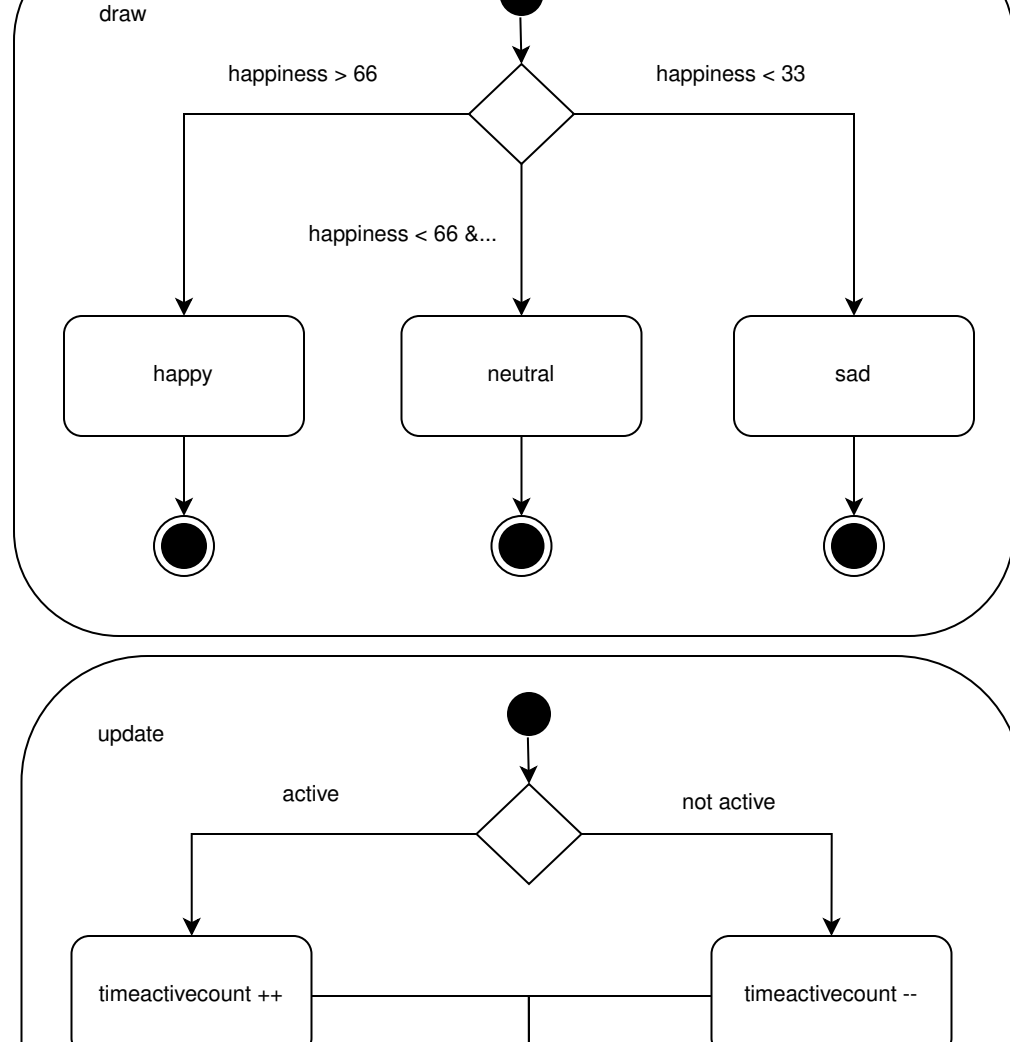
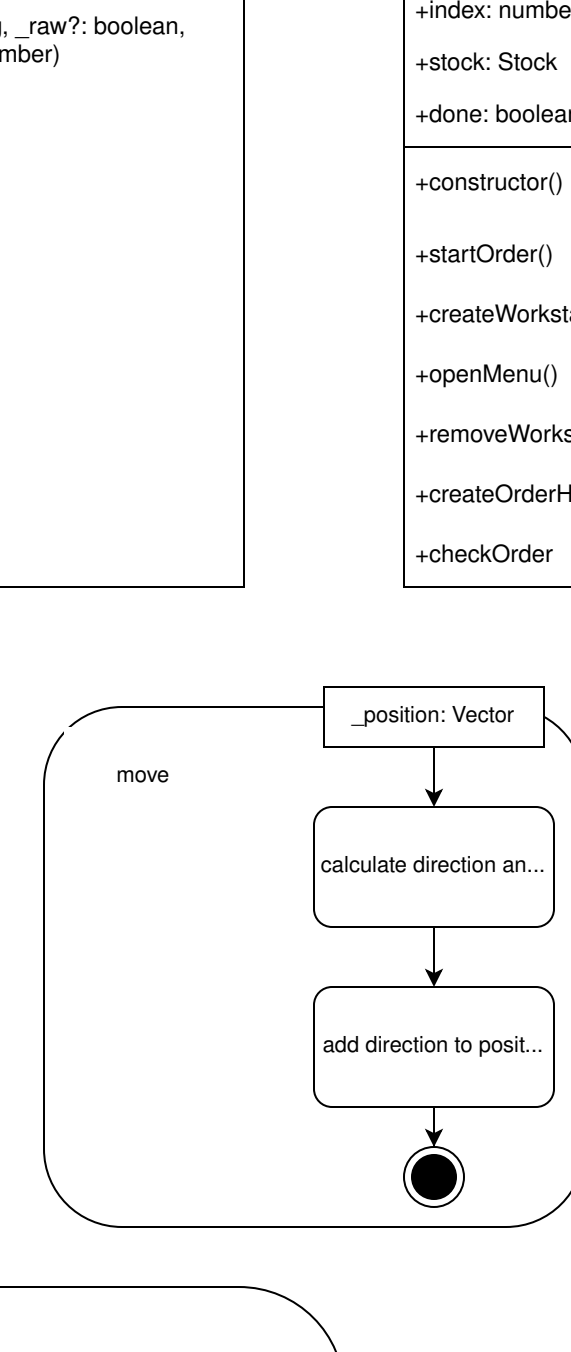
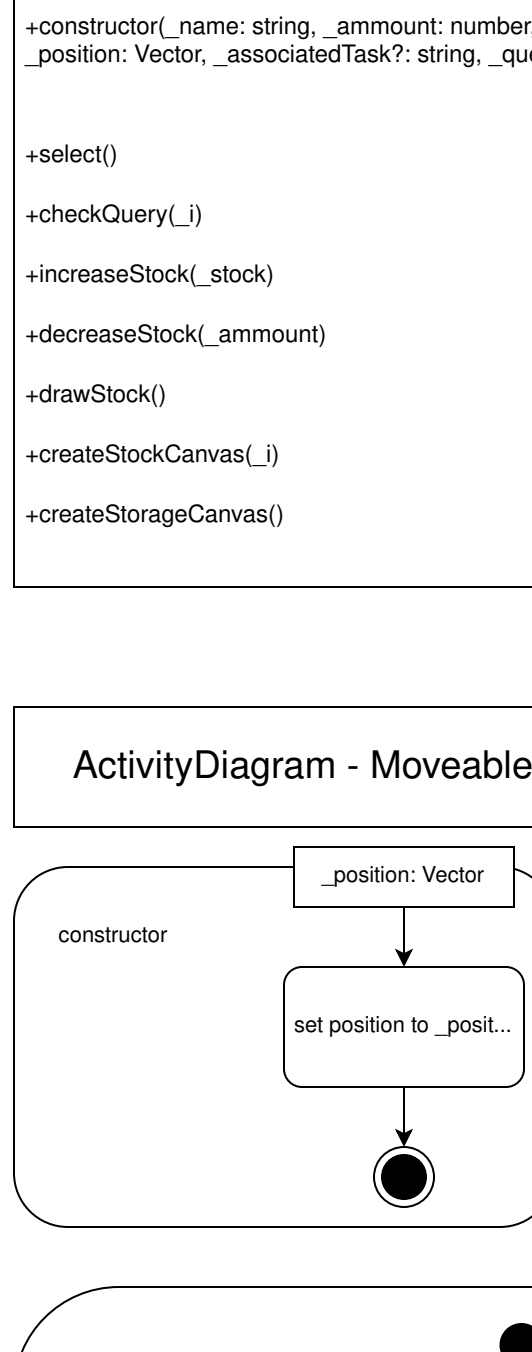
OK



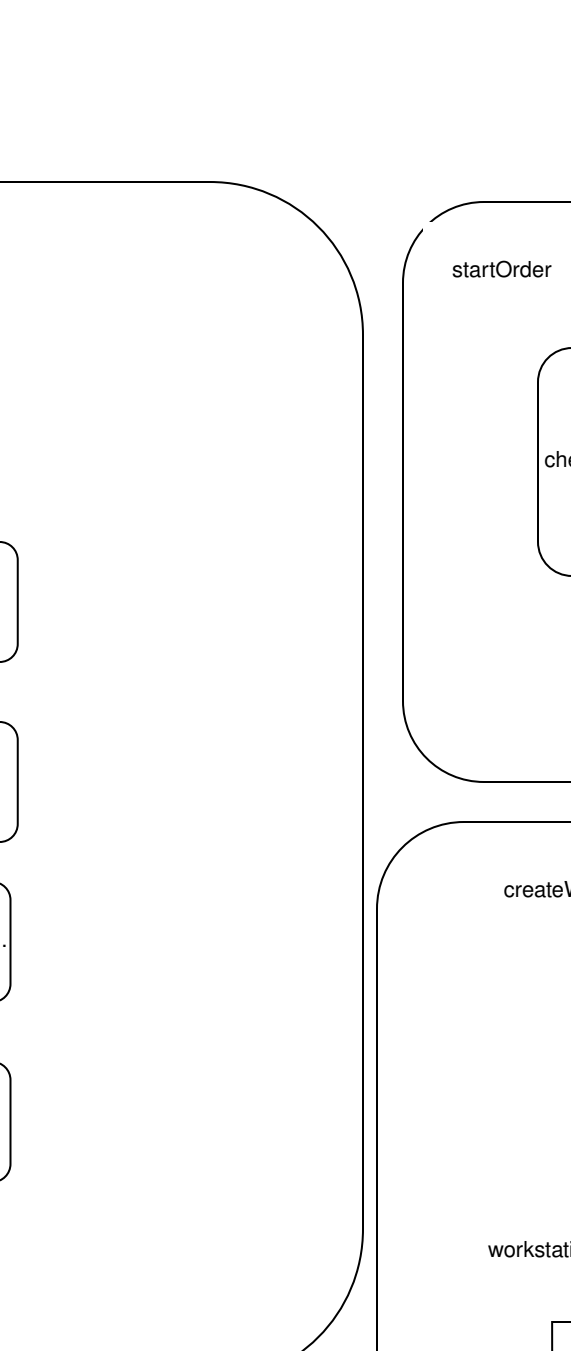
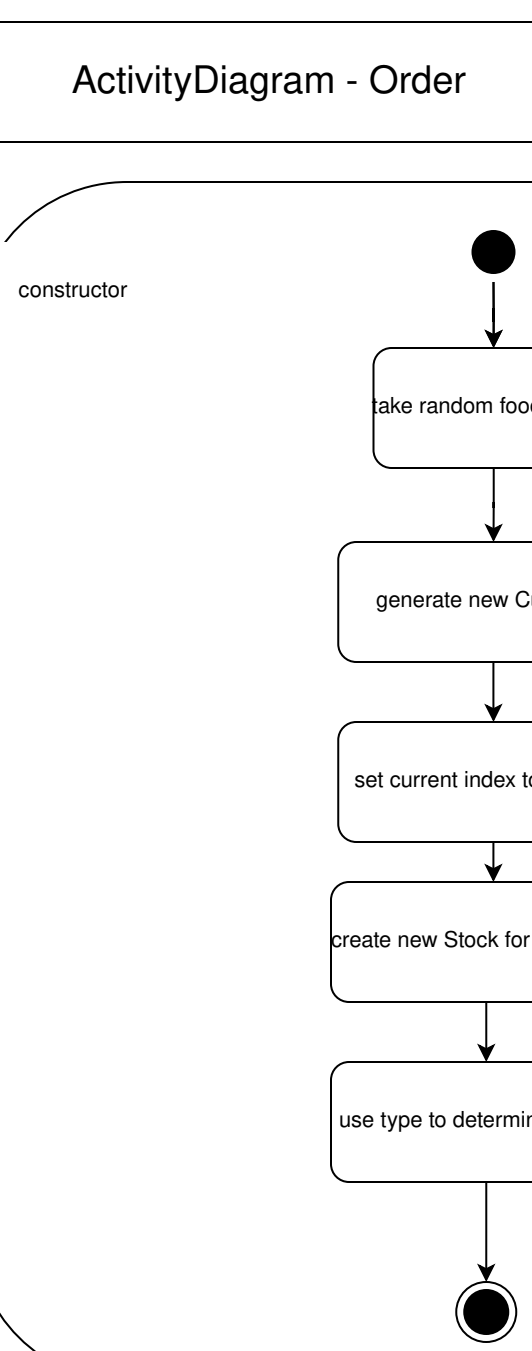
### Class Diagram



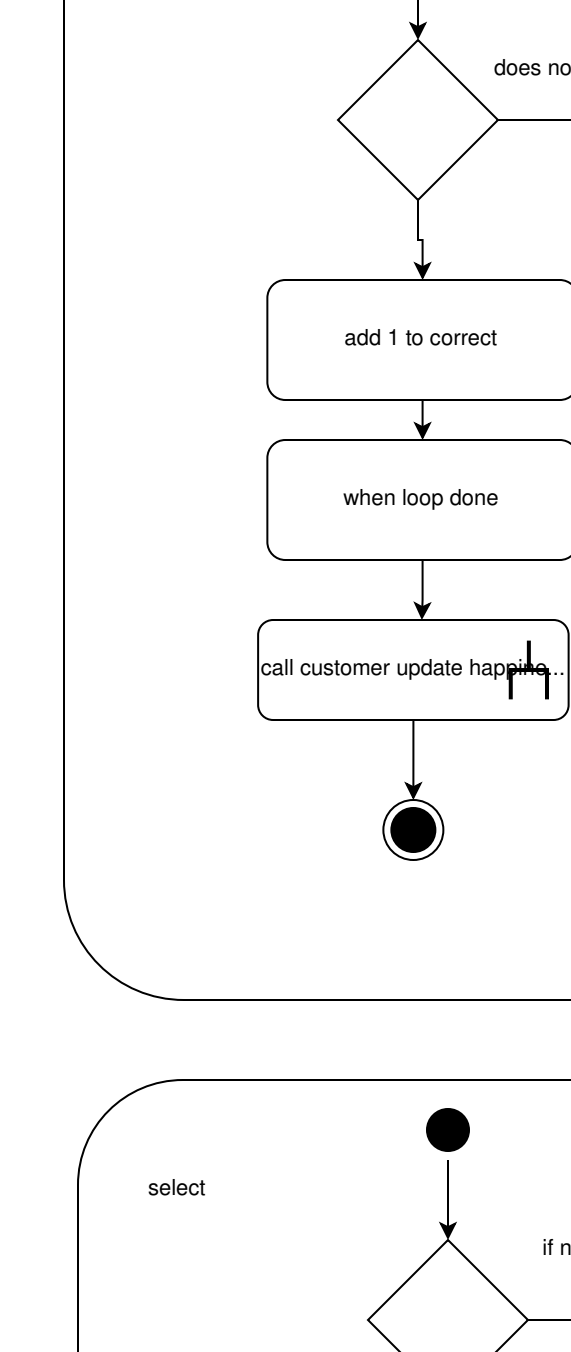
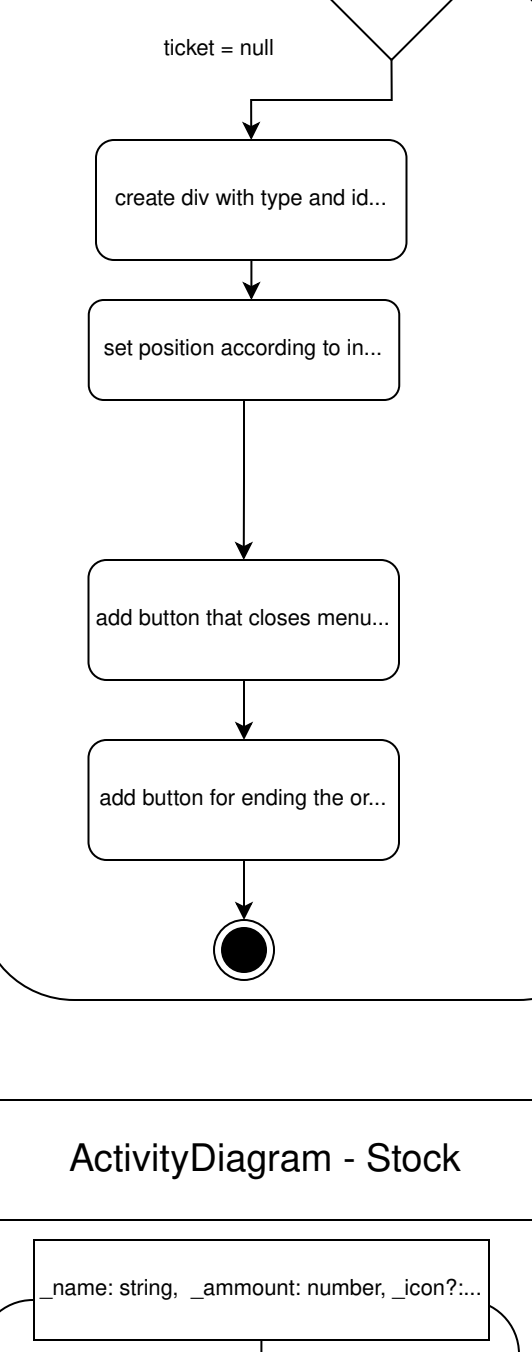
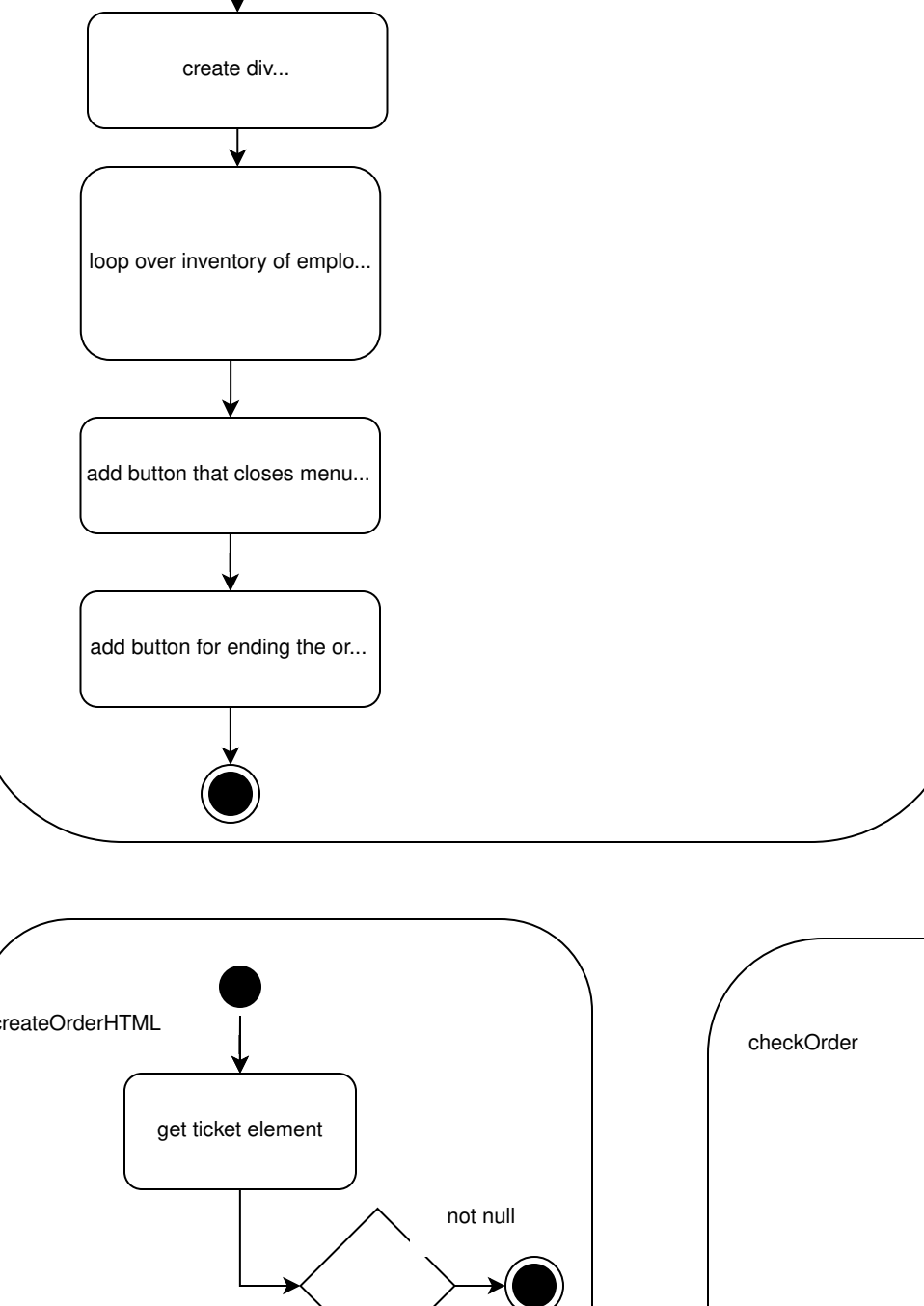
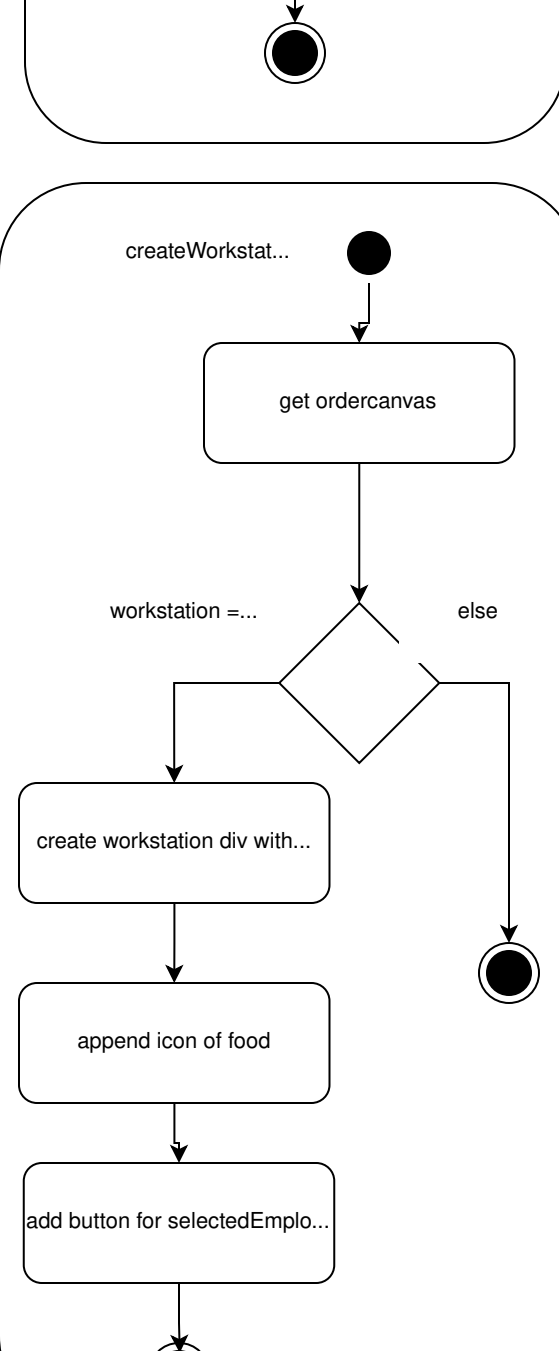
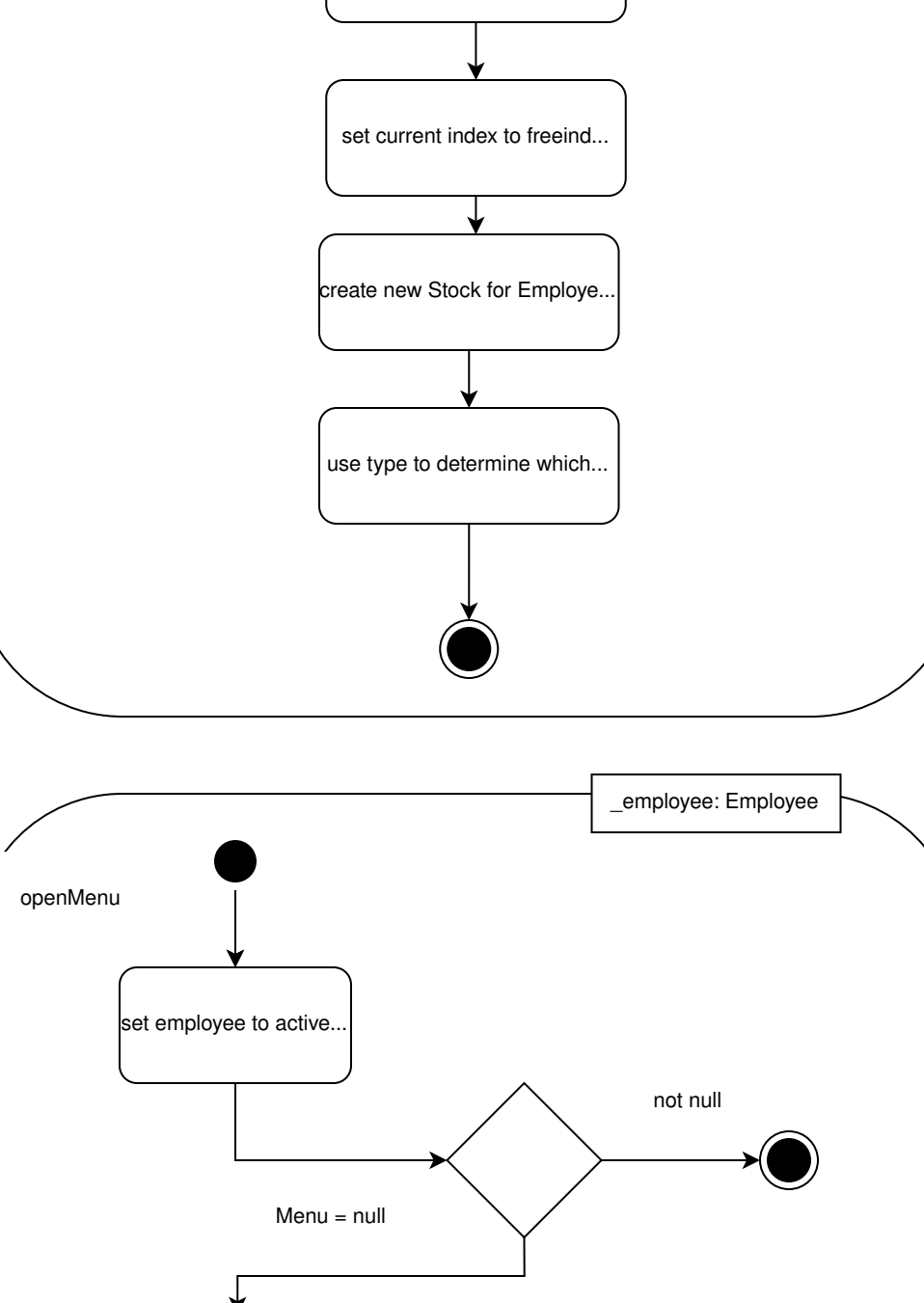
```
constructor(_name: string, _ammount: number,
            position: Vector, _associatedTask?: string, _q
```



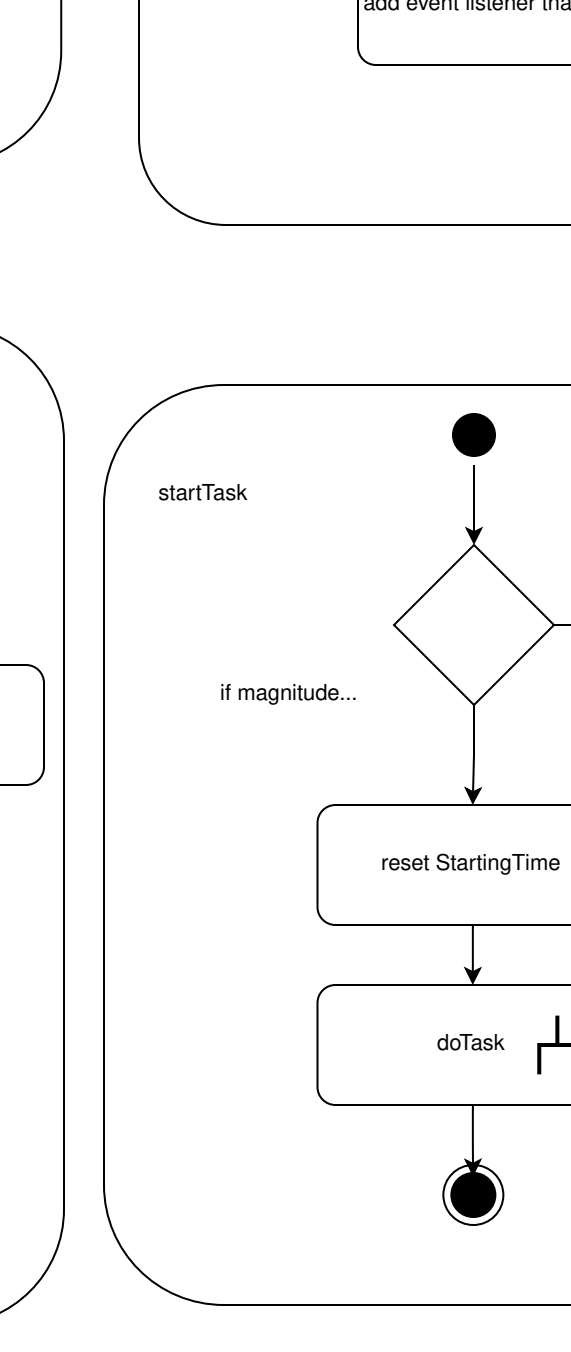
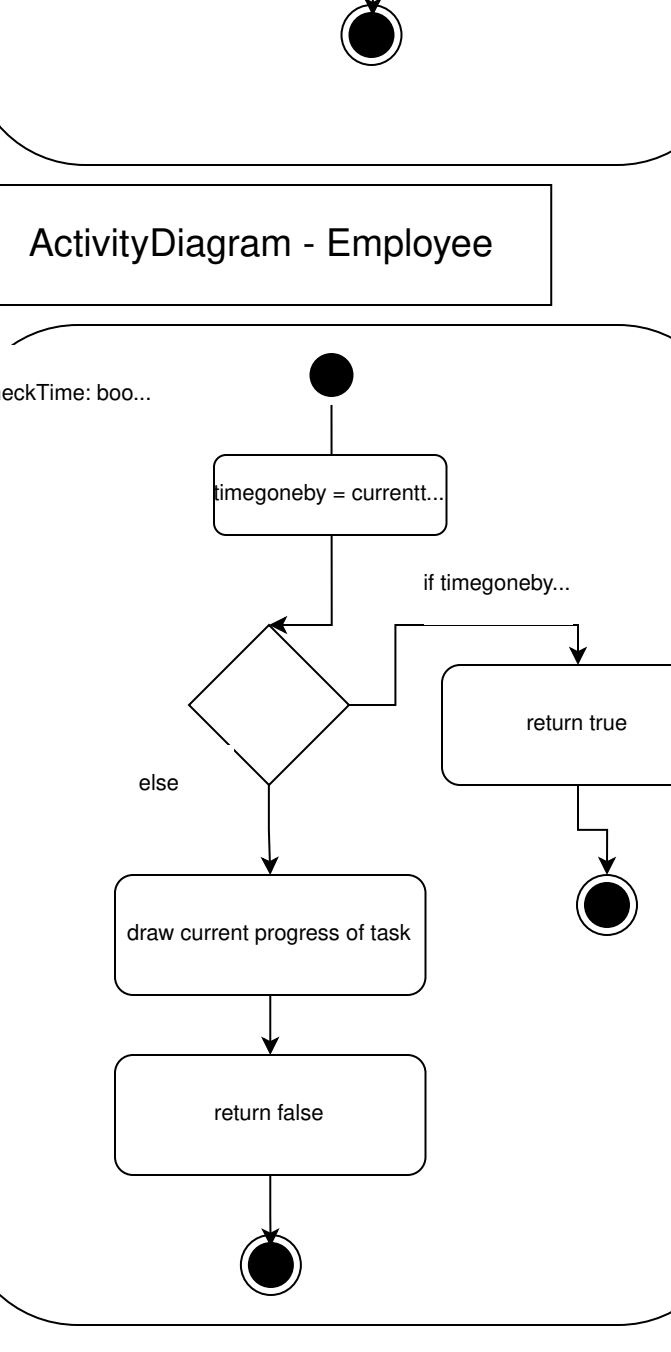
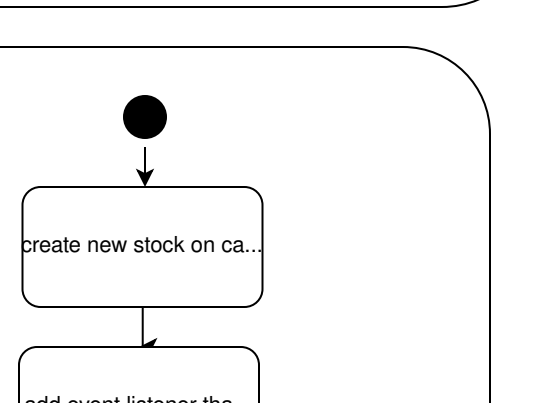
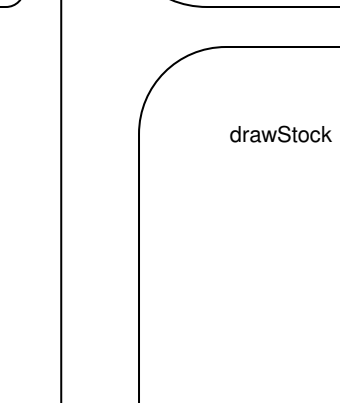
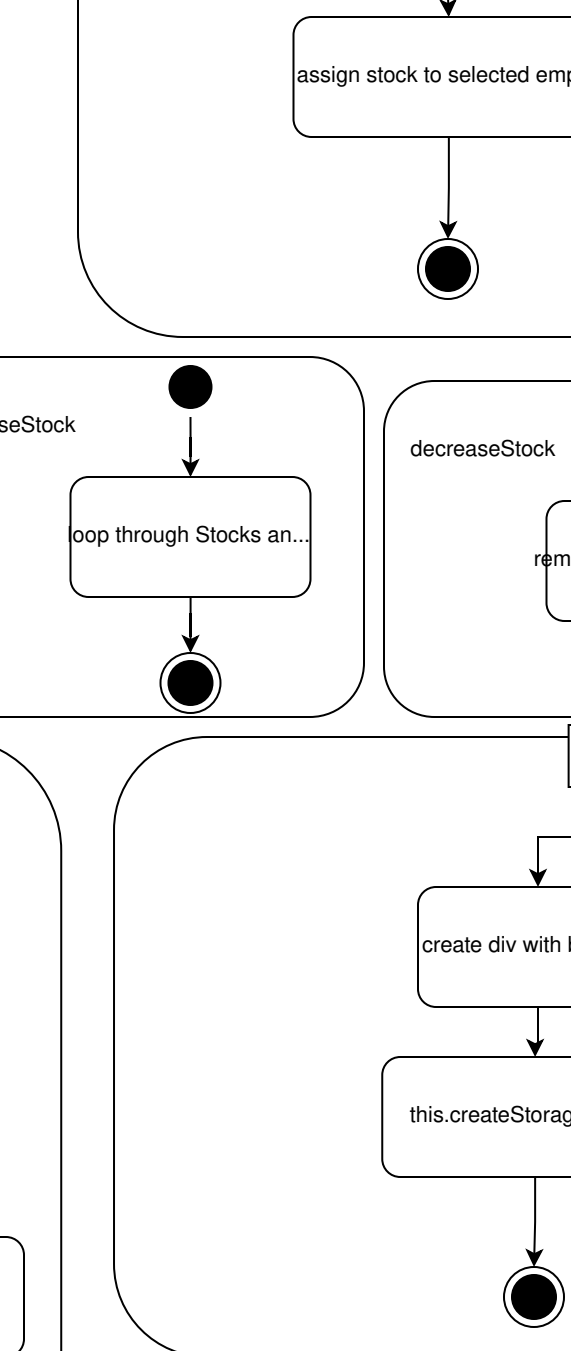
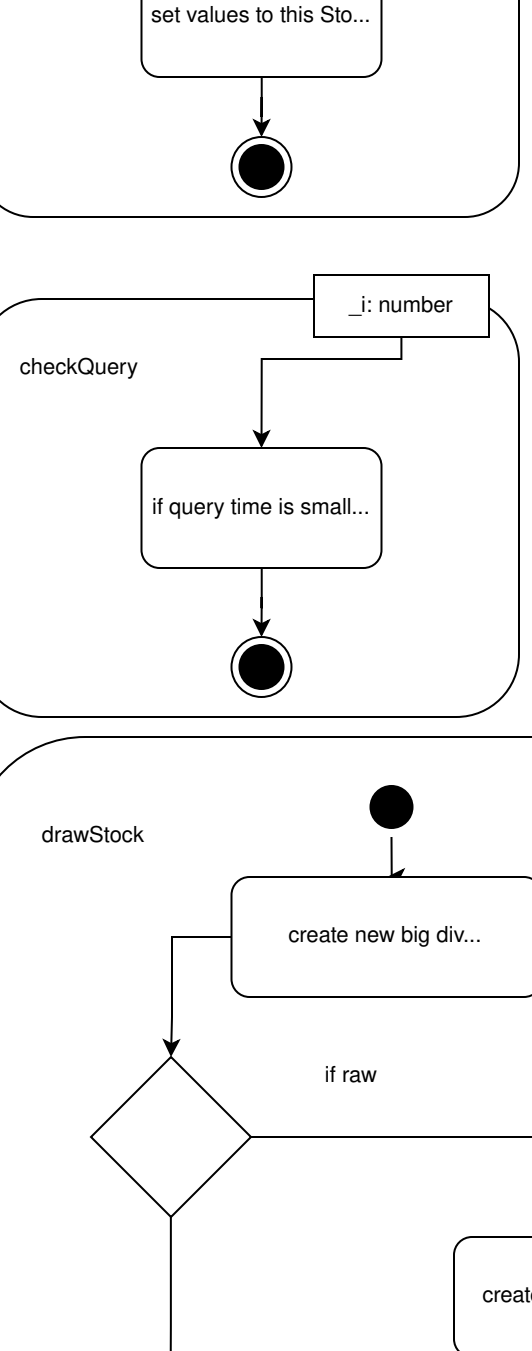
## ActivityDiagram - Order



11



set values to this Sto...



## doTask

