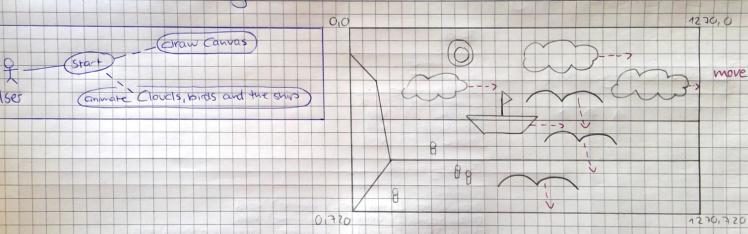
Beach Classs: Use-Case-Diagram

Beach Classes: VI-Scribble



Beach Classes: Class Diagramy Bird Canvas Rendering Context x: number; y number: speed number: constructor() update (): void diawBrash (): void move () void Background constructor(); dawwater (x number - y number) void draw Sand (_x wase, _y woulder) void drow Sky (x: number - y: number - stroke (dor string) wold draw sun (x:number - y number - strake (olor: string - fillolor string) · void Chaw Person (): void daw Bay (-x, home -y homet, fill dor string) void Cloud xlumber y : wumber Speed number draw Cloyd 1(1) void Hraw Cloud 2(): void MOVETOWARD 1 void Ship X: number y: number drawship 1 (): void move Forward 1 void

Reach Classes: Activity Diagram - Main draw Clouds load letienumber=0 [ic clauds length] install Load-Listener nandleLoad (drawClovery Kitt) Let Crc2: Canvas Rendering Context 20 let canvas: HTML Canvas Element. draw Ship let imgData: Image Data; animate let big Clouds Cloud []= [] leti: number = 0 Metinumber = 0 let small (louds - Cloud [] = []. kt ShipBackground: Ship[]= [] Thir Backgroud length (0) [icho Clarce length let flying Bird Bird []=[]; I Smalklocks engly let 2: number = 10; (auds move) dowship() forward (itt) Lyfclads reach end of canvas; go back to start draw Bird 0 hardelood - event Event 1 draw Clouds () th let i number = 0 cawas = document. get Elements by kg Name ("canvas") [] [flying 3 irds length] Gor-loop ship as obove crc2 = canvas ge+Contex+ (-2d-) draw Birds)) let bG Background new Background (draw Ship () m) Create new shirtbird & Clouds with (fo-loop bird as obove) for-100p (drawBird()th) set lime out (animate, 10) >0 L) setTimeout (animate 10)

Beach Classes: Activity Diagram - Background -x nombre vumber x: number - strake Color string drawSound - y: number draw Syn | Bucolog string constructor . let gradient Canvas Gradient let 11: number = 40 (this draw sky let 2: unuver = 300 let gradient: CanvasGradient make gradient (this drawbater (make gradient (fillstyle = gradient) this draw Sand (fillstyle = gradient) (this draw sun (crcl. arc (this draw Person (this craw Bay) -x: number drawsky - Thomas string (draw Person , let gradient Canvas Gradient create circles and X: number drawWater _ y: number squares with cauvas make atadient C102 (111(); let gradient: Canvas Gradient (fill Style - gradient) (make gradient) (fill style = gradient) 0 -x: wunber - GHCOLO string Idraw Bay 0 let gradient: convas Gradient make gradient and (create Bay with line ial) (0) (Stroke() & close Path()

Beach Classes: Activity Diagram - Bird X: number y:number speed: number constructor move This x+= Math random () . 1+1 This, draw Birds 1(), this.y+= Maty. condous () . 5+1 for-loop-> if birds leave canvas they come again update Ahis draw Birds 1(); this move (), th

Beach Classes: Activity Diagram - Cloud X: humber v: number speed: number Craw Cloud 1 draw Clard 2 0 Create Cloud with Create Cloud with (circles: crc2.arcl) Girdes: crcz arc() crc2.closefath()& (Crc2. close Path ()2) (crcz. fill) ercz fill) movetorward This. X+= +his. speed - (+0.5)

Beach Classes: Activity Diagram - Ship

X: number y number speal: number

Create triangle, stick and body of the ship

This. x+=4his speed · (+1.5)