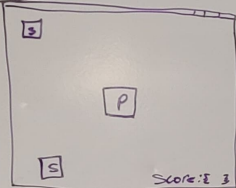


Rundown



Sprites

Player

Collectables

Features

1. Audio
 - ↳ Sound Files
 - ↳ Audio Management
 - ↳ Loading Files
 - ↳ Playing/Stopping Audio
2. Graphics
 - ↳ Sprite Files
 - ↳ Graphics Management
 - ↳ Loading Files
 - ↳ Rendering Files
3. Collision
 - ↳ Rect-Rect Collision
 - ↳ Window Collision
4. Player Movement
 - ↳ Position Motion
 - ↳ Velocity Incorporation
 - ↳ Variable Variability
5. Collectable Management
 - ↳ Score

File Structure

WB-Callbrøp1

Game.cs

Graphics Ref
Audio Ref
Collectables Ref
Player Ref

Setup()

↳ Graphics.Setup()
↳ Audio.Setup()
↳ Setup Collectables
↳ Setup Player

Update()

Collision()

↳ Collectables().Render()
↳ Player.Update()
↳ Player.Render()
↳ Score.Update()
↳ Score.Render()

Collision()

↳ If Player Collides collectable
Score++
Disable Sprite
Move Sprite
Change sprite
Enable sprite

Player.cs

V2 Position
V2 Velocity
T2D Sprite
float Speed

Player()

↳ Position = NV2
↳ Velocity = NV2
↳ Sprite = S.Player

HandleInput()

↳ If K.A:
Vel.x = -1 * Speed
↳ If K.D:
Vel.x = 1 * Speed
↳ If K.W:
Vel.y = -1 * Speed
↳ If K.S:
Vel.y = 1 * Speed

Update()

↳ Position += Velocity

Render()

↳ Graphics.Draw(Sprite, Position)

S.

Sprites.cs

T2D Player
T2D background
T2D wallTexture
T2D Collectable1
T2D Collectable2

Sprites()

↳ Player = G.LT(sprite)
↳ background = G.LT(sprite)
↳ wallTexture = G.LT(sprite)
↳ Collectable1 = G.LT(sprite)
↳ Collectable2 = G.LT(sprite)

So.

Sounds.cs

Music Level/Music
Music win/Music
Music fail/Music
Sound PickupSound
float MusicVolume = 0.8f
float SoundVolume = 0.4f

Sounds()

↳ Level/Music = ALM(sprite)
↳ Win/Music = ALM(sprite)
↳ Fail/Music = ALM(sprite)
↳ PickupSound = ALS(sprite)
↳ ASV(Level/Music, MusicVolume)
↳ ASV(Win/Music, MusicVolume)
↳ ASV(Fail/Music, MusicVolume)
↳ ASV(PickupSound, volume)

Collectable.cs

↳ V2 Position
↳ T2D Sprite
Sound PickupSound
bool Visible = true

Collectable()

↳ Position = NV2
↳ Sprite = S.Collectable1
↳ PickupSound = So.PickupSound

Render()

↳ If visible
G.Draw(Sprite, Position)

Enable()

↳ Visible = true

Disable()

↳ Visible = false

Move()

↳ Position = Random Position

SwapSprite()

↳ Sprite = S.Collectable/sprite

Project Structure

Placeholder: Team Name
Collection: Game Name

Team-Placeholder-A4-Collection

↳ GameIcons
↳ Assets
↳ Sounds
↳ Graphics
↳ Game.cs
↳ Collectables.cs
↳ Player.cs
↳ Sounds.cs
↳ Sprites.cs

Process-Work

↳ Name.pdf
↳ ETL

Ignore

README

↳ Instructions
↳ Team member names

Scene Helper

int CurrentScene

if CS = 0:

Title

CS = 1:

Intro

CS = 2:

Level 1

CS = 3:

Level 2

CS = 4:

Fail

CS = 5:

Win

Play on
Above indiv
Level



Level

BbSprite

ScoreToPass

Type of Collectables

MUSIC

L Fail

L Win

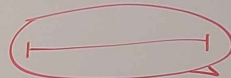
Text Bubble

Helper.TextBubble()

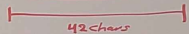
Params:

String: ScreenPos

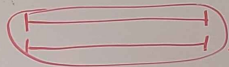
String: text to print



#Lines = 1



= 2



Should scale
with # Lines

If (text <= 42 chars):

Print 42 char on Line 1

Else if (text > 42 && < 84):

Print 42c on Line 1

Print 42c on Line 2

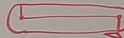
etc.

Else if (text > 84):

Split to another

Text bubble underneath

Bubble:



Capable of quadrupling
from edge of
Screen

Trim

Word wrapping

TempString = "Test String with Spaces"

1. Find all spaces (Empty spaces)
2. Find nearest - to HZ/84 respectively



is

11c
Hello world

Hallo world
d

10
+++++