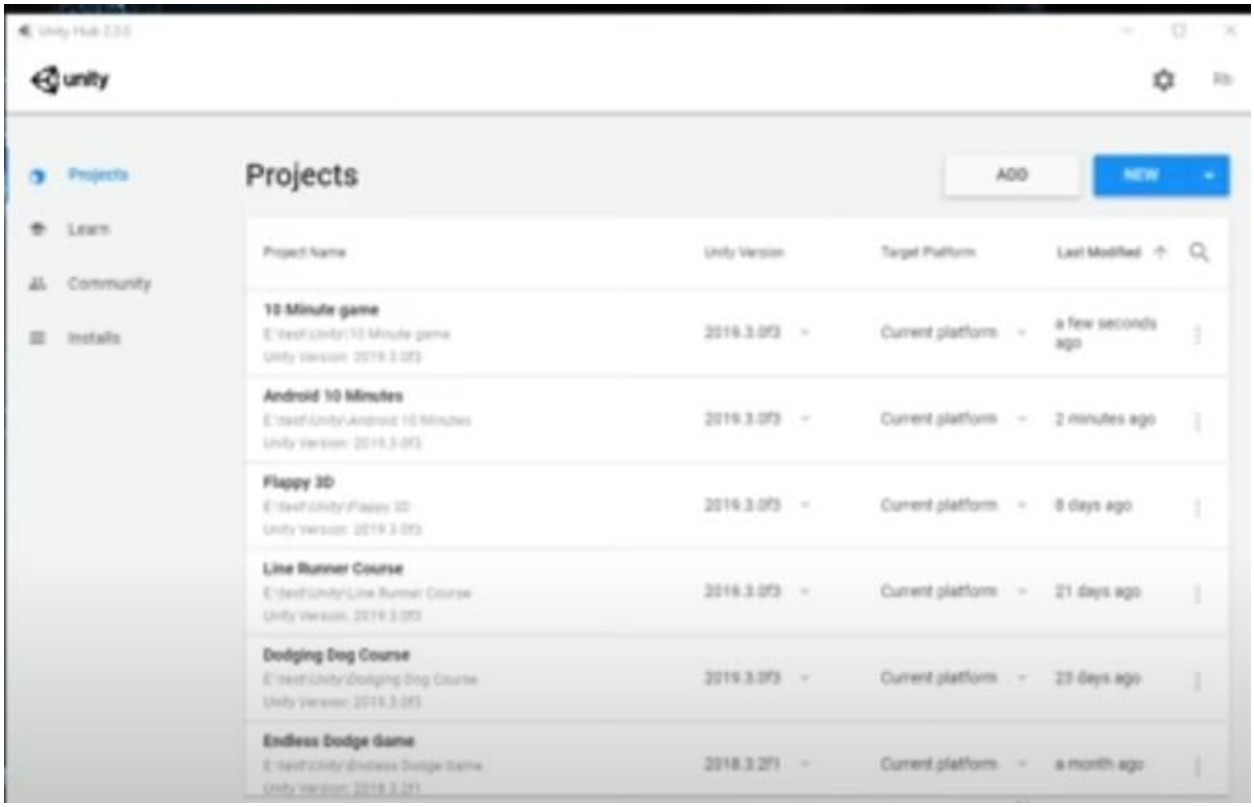
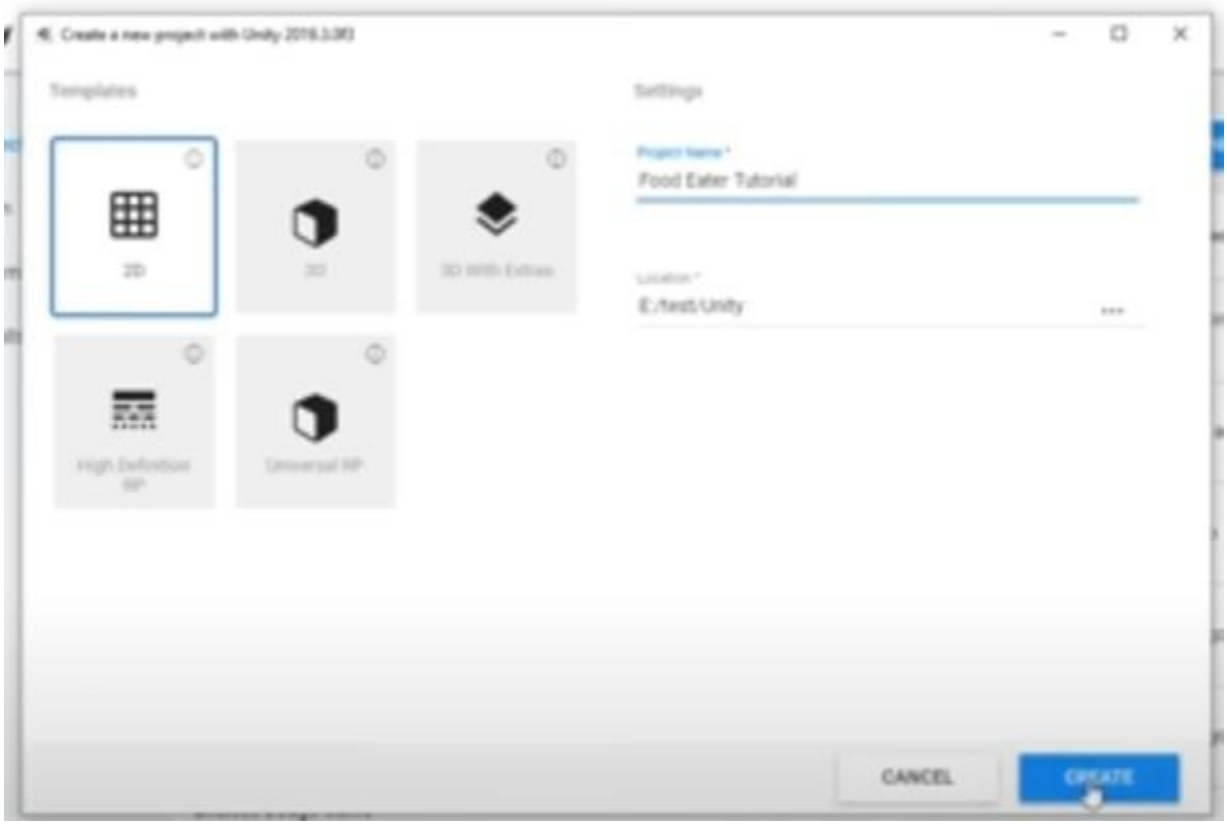
TYCS SEM 5

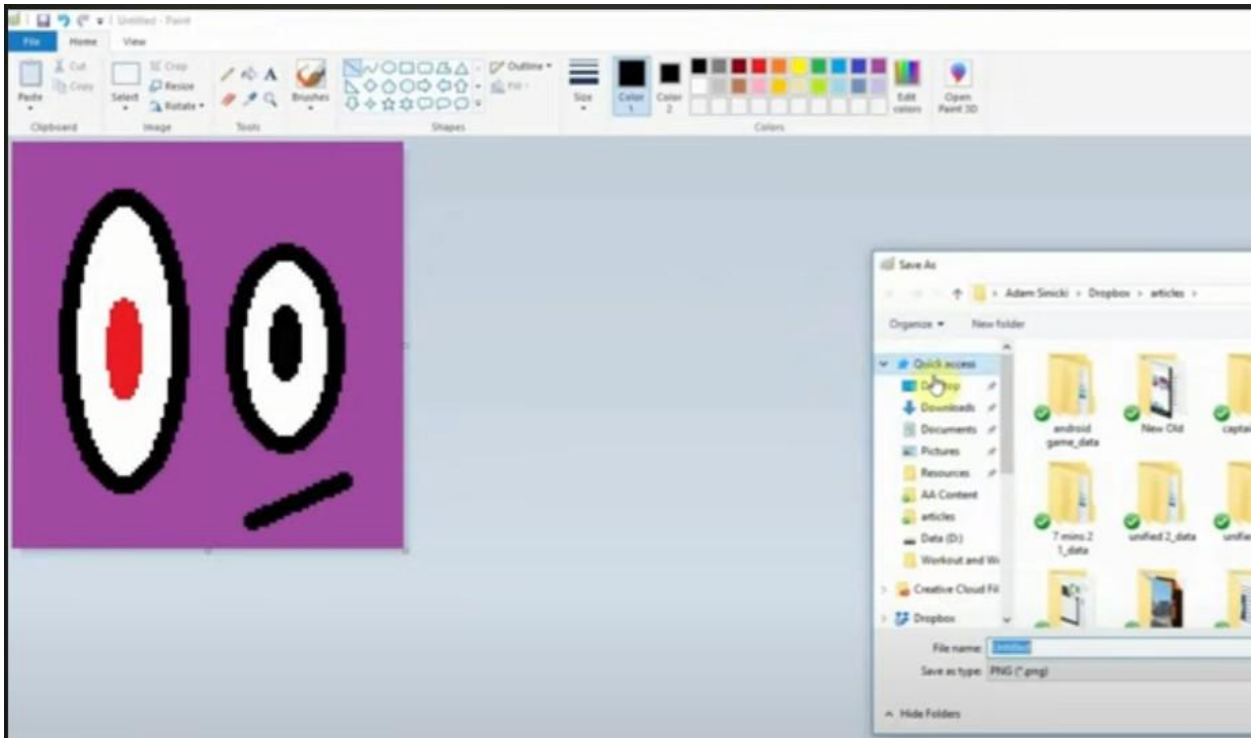
GAME PROGRAMMING

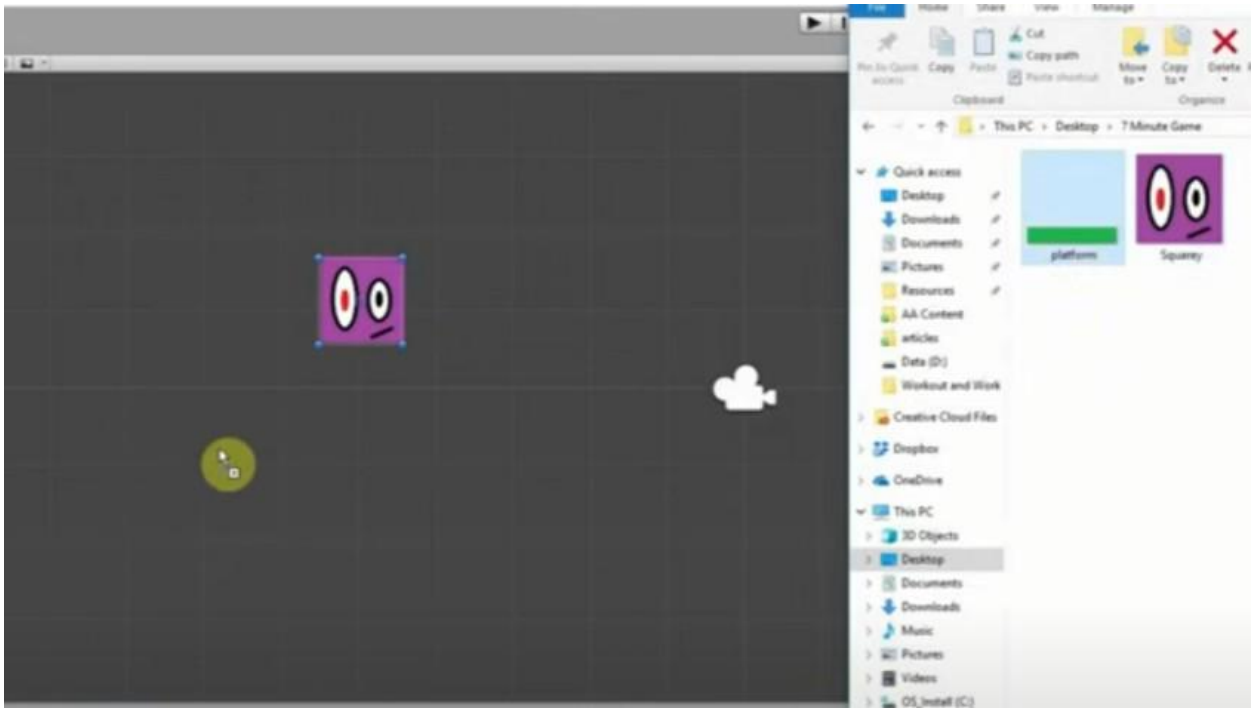
Practical no 9

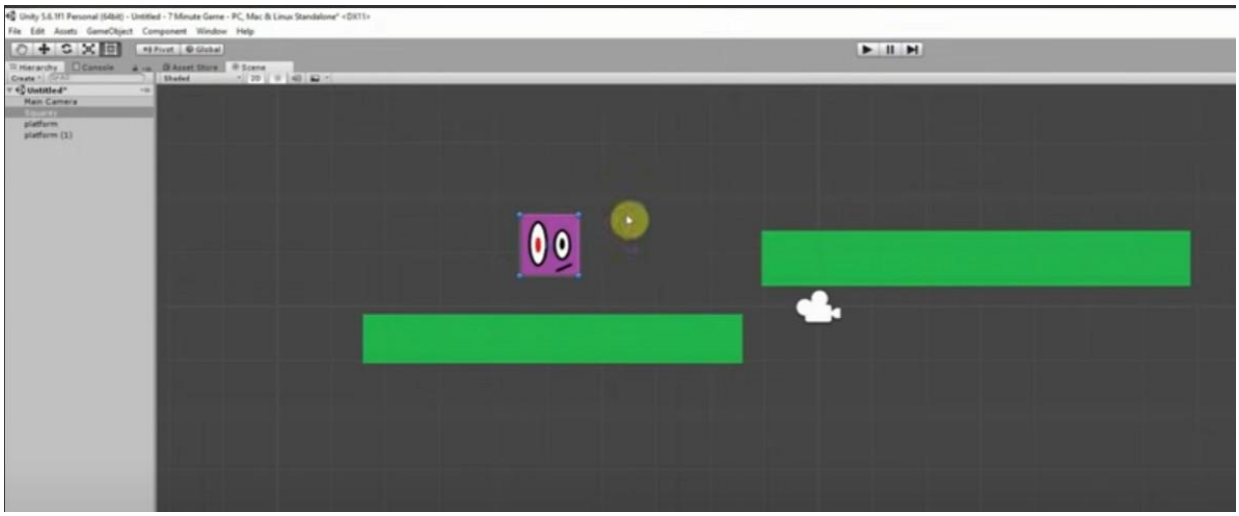
AIM:Develop Android Game with Unity




Templates
Settings
Food Eater Tutorial
E:/test/Unity
CANCEL
CREATE

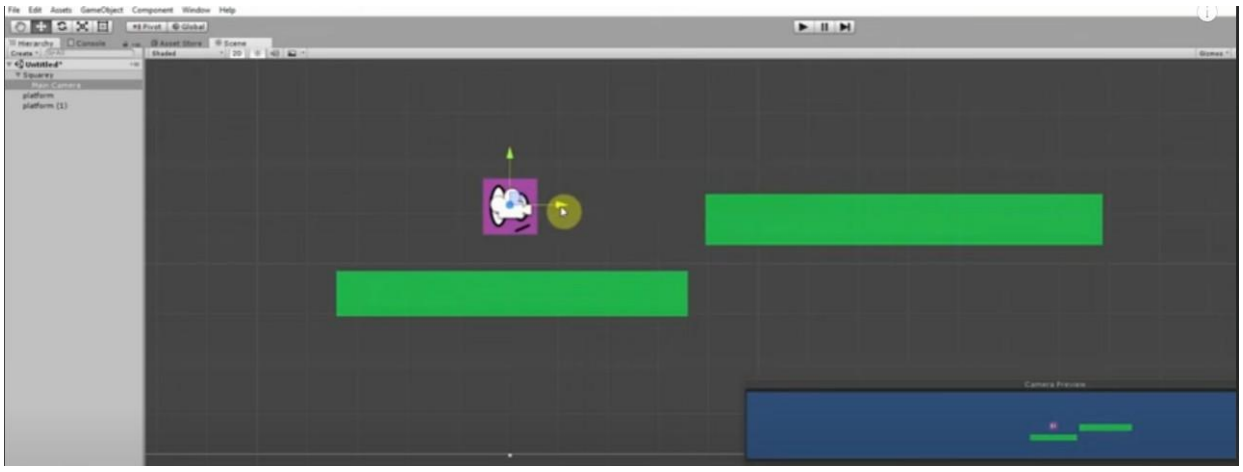



This PC
Desktop
7 Minute Game
Quick access
Desktop
Downloads
Documents
Pictures
Resources
AA Content
articles
Data (D:)
Workout and Work
Creative Cloud Files
Dropbox
OneDrive
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
platform
Squarey

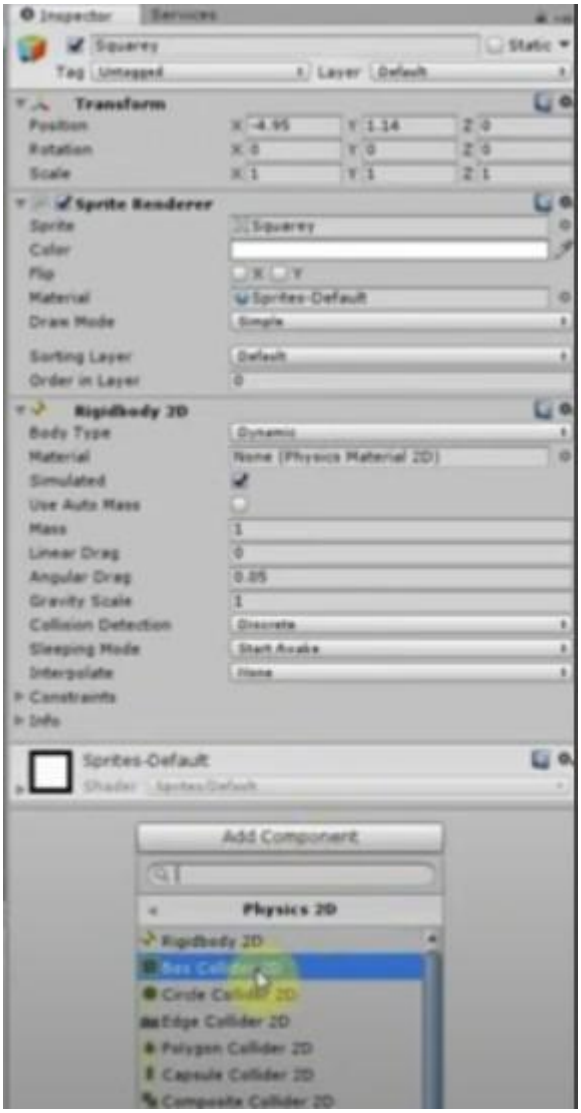



Inspector
Services
Squarey
Static
Tag
Untagged
Layer
Default
Transform
Position
Rotation
Scale
Sprite Renderer
Sprite
Color
Flip
Material
Sprites-Default
Draw Mode
Simple
Sorting Layer
Order in Layer
Add Component
Component
Mesh
Effects
Physics
Physics 2D
Navigation
Audio
Video
Rendering
Layout
Miscellaneous
Analytics
Scripts
Event


Inspector
Services
Squarey
Static
Tag
Untagged
Layer
Default
Transform
Position
Rotation
Scale
Sprite Renderer
Sprite
Color
Flip
Material
Sprites-Default
Draw Mode
Simple
Sorting Layer
Order in Layer
Add Component
Physics 2D
Box Collider 2D
Circle Collider 2D
Edge Collider 2D
Polygon Collider 2D
Capsule Collider 2D
Composite Collider 2D
Distance Joint 2D
Fixed Joint 2D
Friction Joint 2D
Hinge Joint 2D
Relative Joint 2D
Slider Joint 2D




Inspector
Services
Squarey
Static
Tag
Untagged
Layer
Default
Transform
Position
Rotation
Scale
Sprite Renderer
Sprite
Squarey
Color
Flip
Material
Sprites-Default
Draw Mode
Simple
Sorting Layer
Default
Order in Layer
Rigidbody 2D
Body Type
Dynamic
Material
None (Physics Material 2D)
Simulated
Use Auto Mass
Mass
Linear Drag
Angular Drag
Gravity Scale
Collision Detection
Discrete
Sleeping Mode
Start Awake
Interpolate
None
Constraints
Info
Sprites-Default
Shader
Add Component
Component
Mesh
Effects
Physics
Physics 2D
Navigation
Audio
Video


Inspector
Services
Squarey
Static
Tag
Untagged
Layer
Default
Transform
Position
Rotation
Scale
Sprite Renderer
Sprite
Squarey
Color
Flip
Material
Sprites-Default
Draw Mode
Simple
Sorting Layer
Default
Order in Layer
Rigidbody 2D
Body Type
Dynamic
Material
None (Physics Material 2D)
Simulated
Use Auto Mass
Mass
Linear Drag
Angular Drag
Gravity Scale
Collision Detection
Discrete
Sleeping Mode
Start Awake
Interpolate
None
Constraints
Info
Sprites-Default
Shader
Add Component
Physics 2D
Rigidbody 2D
Box Collider 2D
Circle Collider 2D
Edge Collider 2D
Polygon Collider 2D
Capsule Collider 2D
Composite Collider 2D

SAME FOR ALL PLATFORM.

ADD THE SCRIPT

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class PlayerControls : MonoBehaviour {
    public Rigidbody2D rb;

    // Use this for initialization
    void Start () {

        rb = GetComponent<Rigidbody2D>();

    }

    // Update is called once per frame
    void Update () {

        rb.velocity = new Vector2(1, rb.velocity.y);
```

```
// Update is called once per frame
void Update () {

    rb.velocity = new Vector2(3, rb.velocity.y);

    if (Input.GetMouseButtonDown(0))
    {
        rb.velocity = new Vector2(rb.velocity.x, 3);
    }
```

```
public class PlayerControls : MonoBehaviour
{
    public Rigidbody2D rb;
    public Transform groundCheck;
    public float groundCheckRadius;
    public LayerMask whatIsGround;
    private bool onGround;
```

```
// Update is called once per frame
void Update()
{

    rb.velocity = new Vector2(3, rb.velocity.y);
    onGround = Physics2D.OverlapCircle(groundCheck.position, groundChec)
```

```
Vector2(3, rb.velocity.y);
2D.OverlapCircle(groundCheck.position, groundCheckRadius, whatIsGround)
```

```
    if (Input.GetMouseButtonDown(0) && onGround)
    {

        rb.velocity = new Vector2(rb.velocity.x, 3);
```

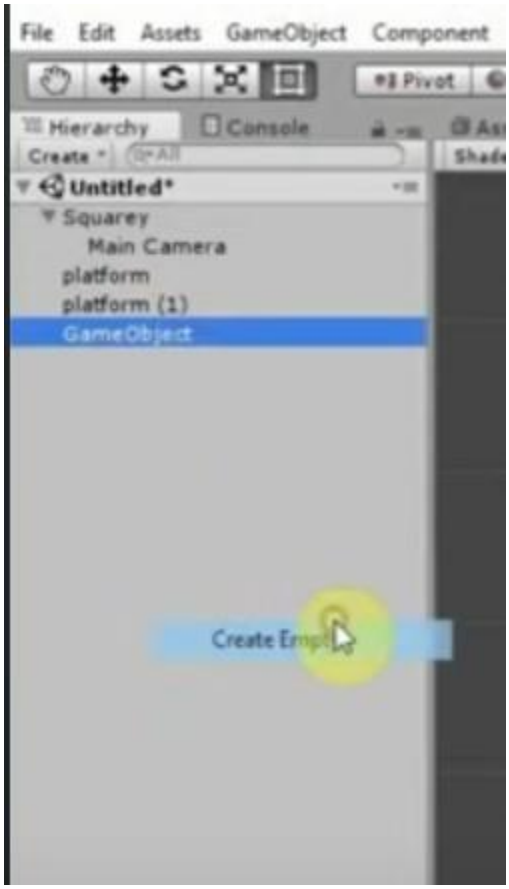
class player

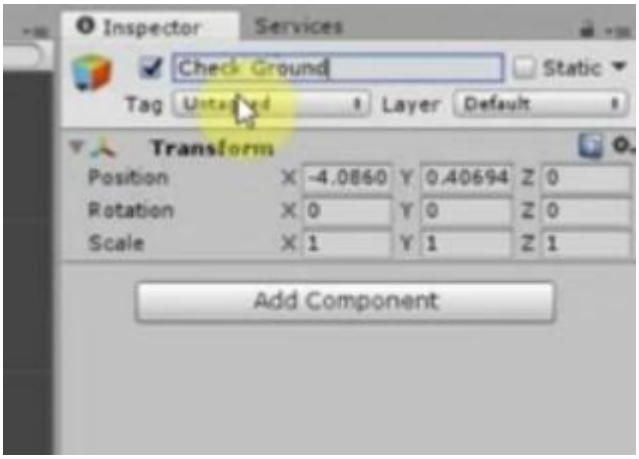
{

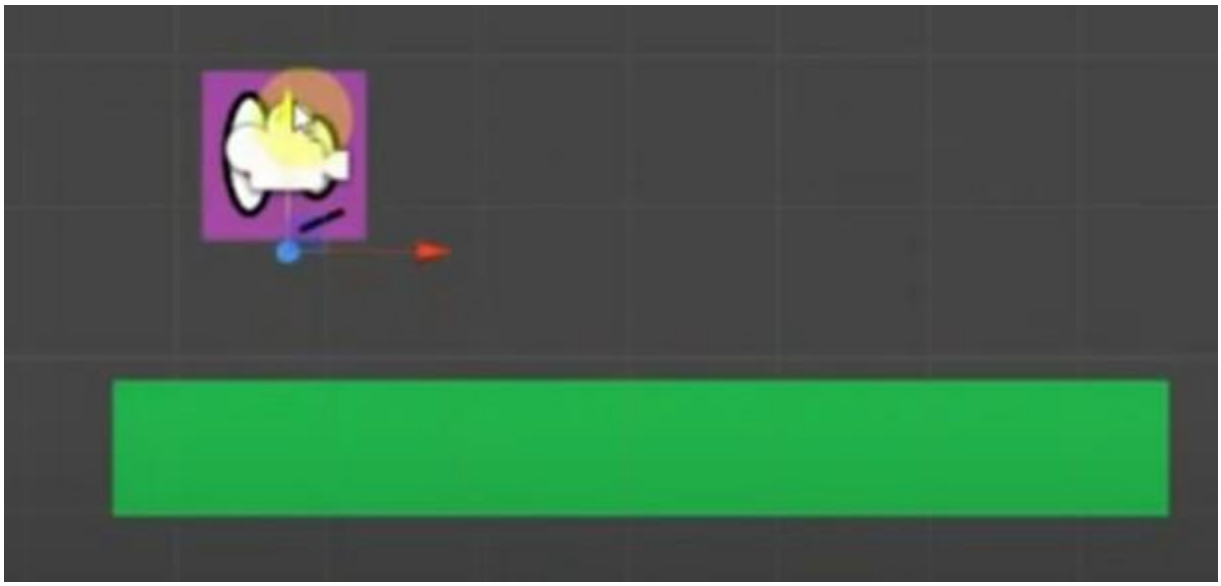
public Rigidbody2D rb;

public Transform groundCheck;

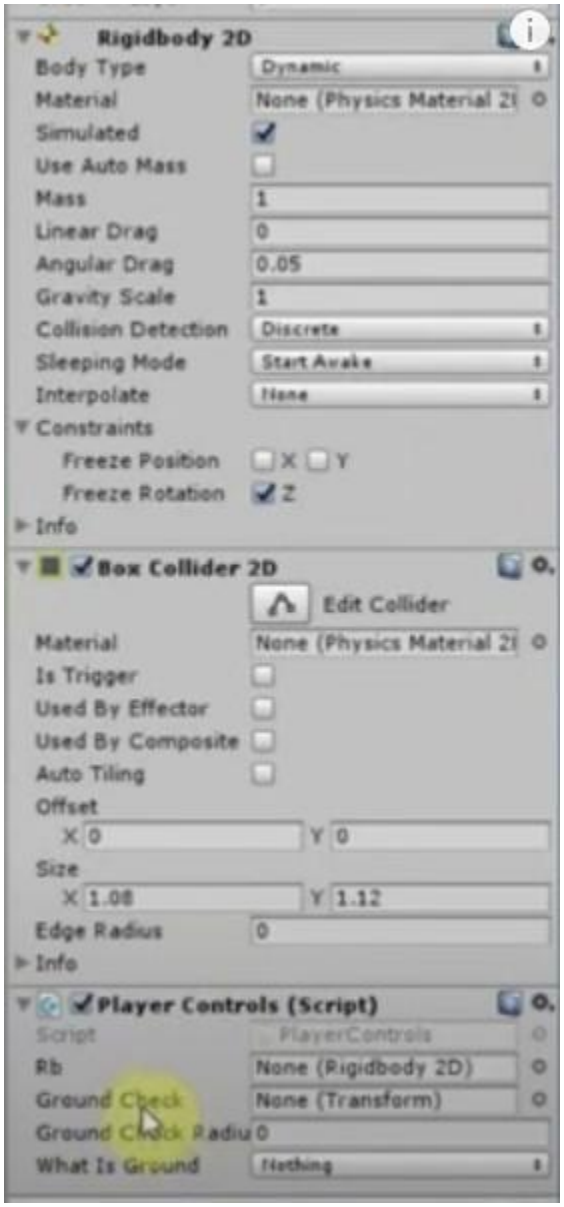
```
public float groundCheckRadius;

public LayerMask whatIsGround;

private bool onGround;

void start()

{

rb=GetComponent<Rigidbody2D>();

}

void update()

{

rb.velocity = new Vector(1,rb.velocity.y);

onGround =Physics2D.OverLapCircle(groundCheck.position
,groundCheckRadius, whatIsGround);

if(Input.GetMouseButton(0) && onGround)

{

rb.velocity = new Vector2(rb.velocity.x , 3);

}
```

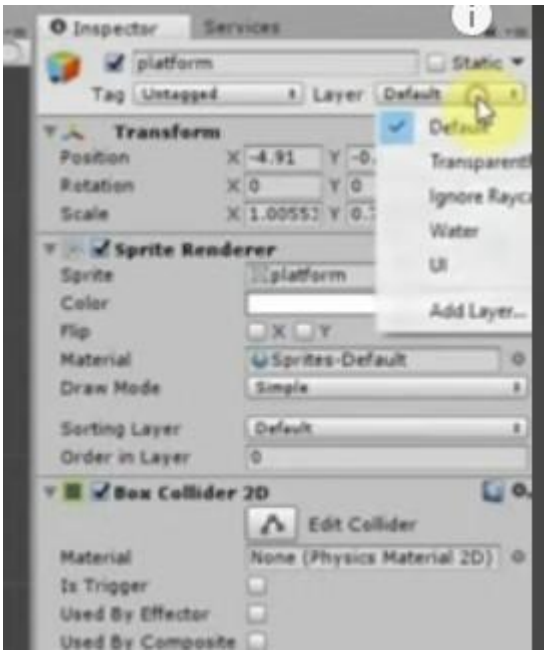


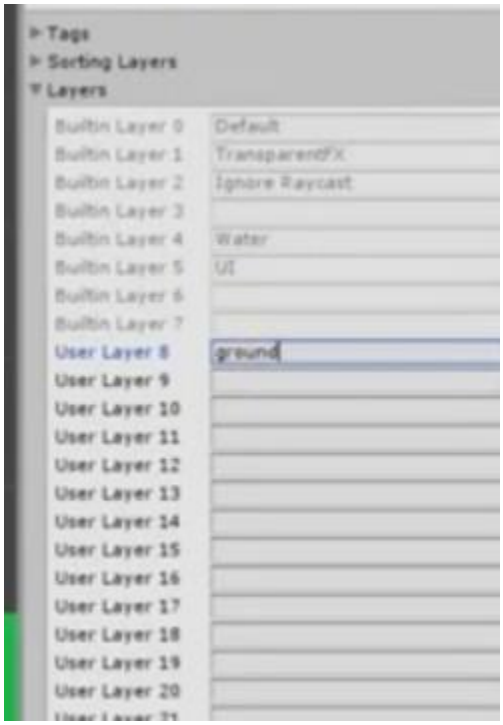




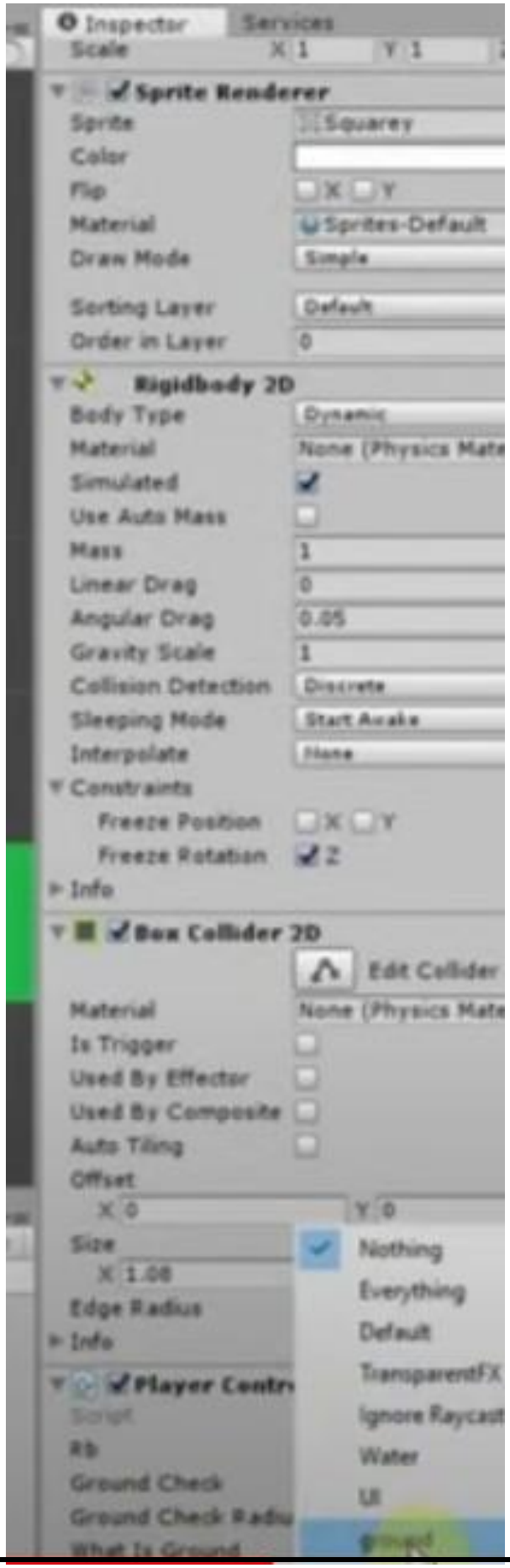
MOVE THE GAME OBJECT TO SQUAREY .

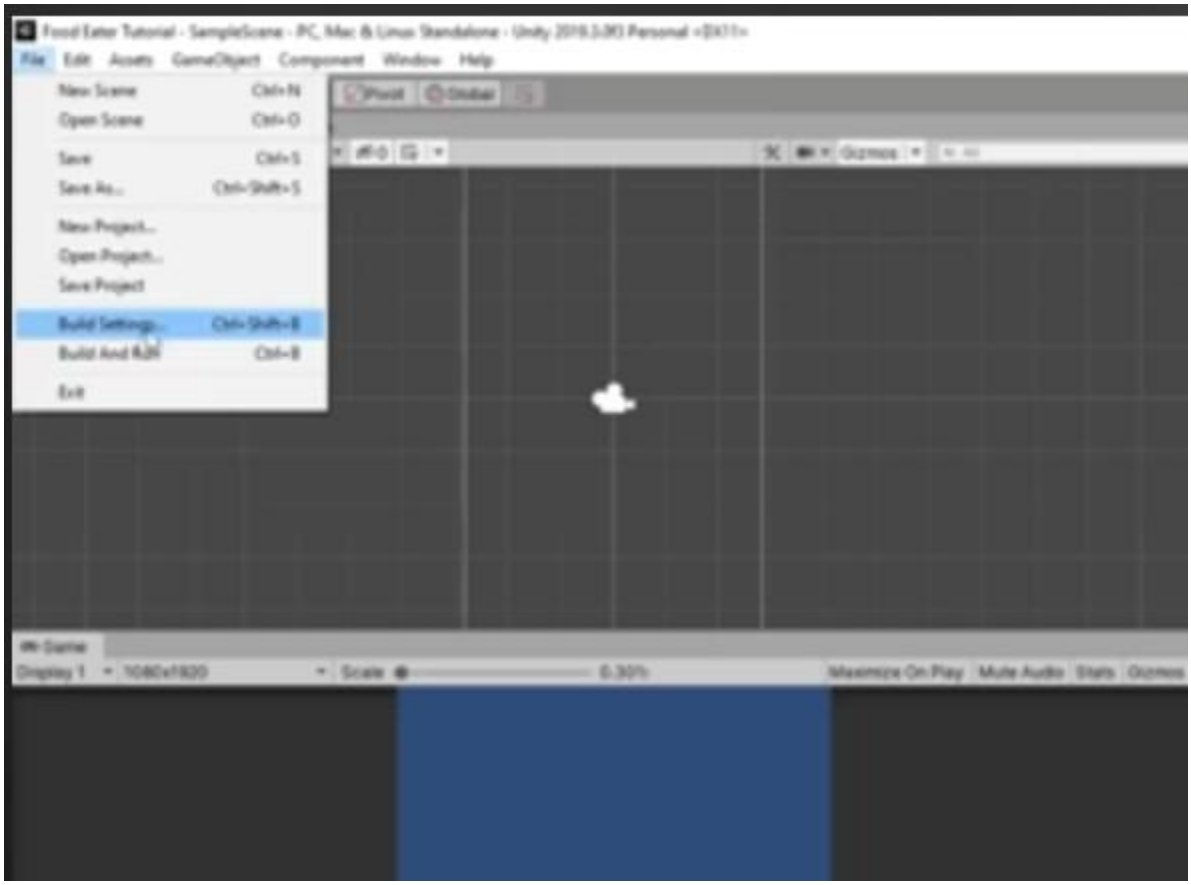

Rigidbody 2D
Body Type
Dynamic
Material
None (Physics Material 2
Simulated
Use Auto Mass
Mass
1
Linear Drag
0
Angular Drag
0.05
Gravity Scale
1
Collision Detection
Discrete
Sleeping Mode
Start Awake
Interpolate
None
Constraints
Freeze Position
X Y
Freeze Rotation
Z
Info
Box Collider 2D
Edit Collider
Material
None (Physics Material 2
Is Trigger
Used By Effector
Used By Composite
Auto Tiling
Offset
X 0
Y 0
Size
X 1.08
Y 1.12
Edge Radius
0
Info
Player Controls (Script)
Script
PlayerControls
Rb
None (Rigidbody 2D)
Ground Check
None (Transform)
Ground Check Radiu
0
What Is Ground
Nothing

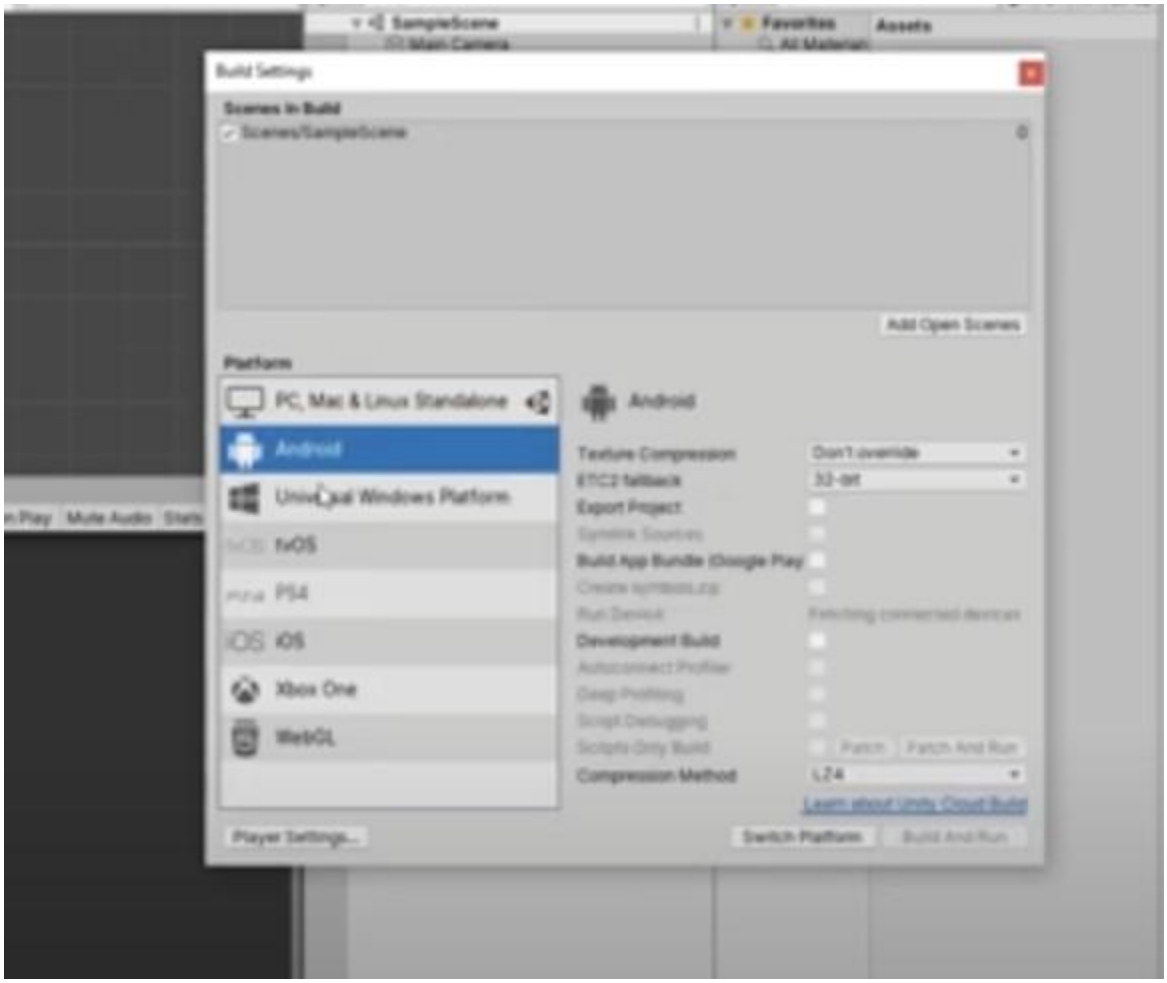

Inspector
Services
platform
Static
Tag
Untagged
Layer
Default
Transform
Position
Rotation
Scale
Sprite Renderer
Sprite
platform
Color
Flip
Material
Sprites-Default
Draw Mode
Simple
Sorting Layer
Default
Order in Layer
0
Box Collider 2D
Edit Collider
Material
None (Physics Material 2D)
Is Trigger
Used By Effector
Used By Composite
Default
Transparent
Ignore Rayc
Water
UI
Add Layer...


Tags
Sorting Layers
Layers
Builtin Layer 0 Default
Builtin Layer 1 TransparentFX
Builtin Layer 2 Ignore Raycast
Builtin Layer 3
Builtin Layer 4 Water
Builtin Layer 5 UI
Builtin Layer 6
Builtin Layer 7
User Layer 8 ground
User Layer 9
User Layer 10
User Layer 11
User Layer 12
User Layer 13
User Layer 14
User Layer 15
User Layer 16
User Layer 17
User Layer 18
User Layer 19
User Layer 20

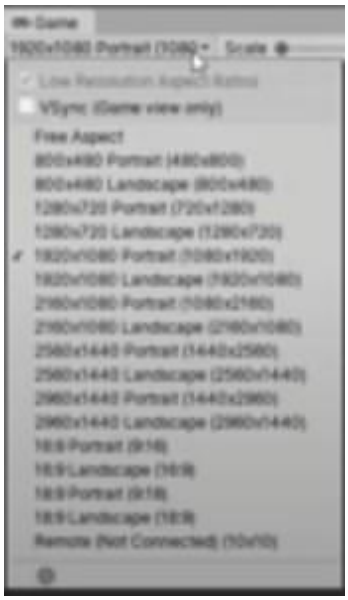

Inspector
Services
platform
Static
Tag Untagged
Layer Default
Transform
Position X -4.91
Rotation X 0
Scale X 1.00553
Sprite Renderer
Sprite platform
Color
Flip X Y
Material Sprites-Def
Draw Mode Simple
Sorting Layer Default
Order in Layer 0
Box Collider 2D
Edit Collider
Material None (Physics Material 2D)
Is Trigger
Used By Effector
Used By Composite
Auto Tiling
Offset X 0 Y 0
Size X 6.97 Y 1.2
Edge Radius 0
Info
Sprites-Default
Shader Sprites/Default
Add Component
Default
Water
UI
Add Layer...

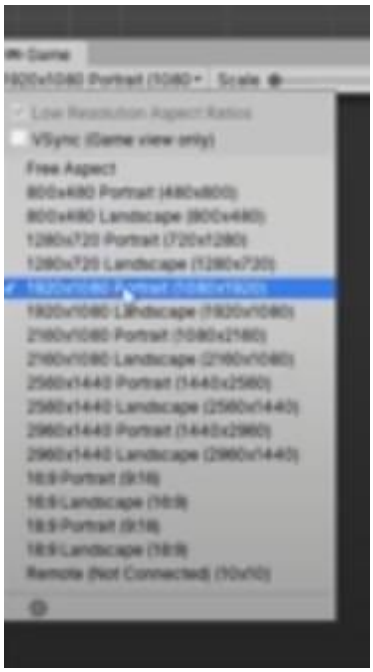

Inspector
Services
Scale
X 1
Y 1
Sprite Renderer
Sprite
Squarey
Color
Flip
X
Y
Material
Sprites-Default
Draw Mode
Simple
Sorting Layer
Default
Order in Layer
0
Rigidbody 2D
Body Type
Dynamic
Material
None (Physics Mate
Simulated
Use Auto Mass
Mass
1
Linear Drag
0
Angular Drag
0.05
Gravity Scale
1
Collision Detection
Discrete
Sleeping Mode
Start Awake
Interpolate
None
Constraints
Freeze Position
X
Y
Freeze Rotation
Z
Info
Box Collider 2D
Edit Collider
Material
None (Physics Mate
Is Trigger
Used By Effector
Used By Composite
Auto Tiling
Offset
X 0
Y 0
Size
X 1.08
Edge Radius
Info
Player Contr
Script
Rb
Ground Check
Ground Check Radiu
What Is Ground
Nothing
Everything
Default
TransparentFX
Ignore Raycast
Water
UI
Ground

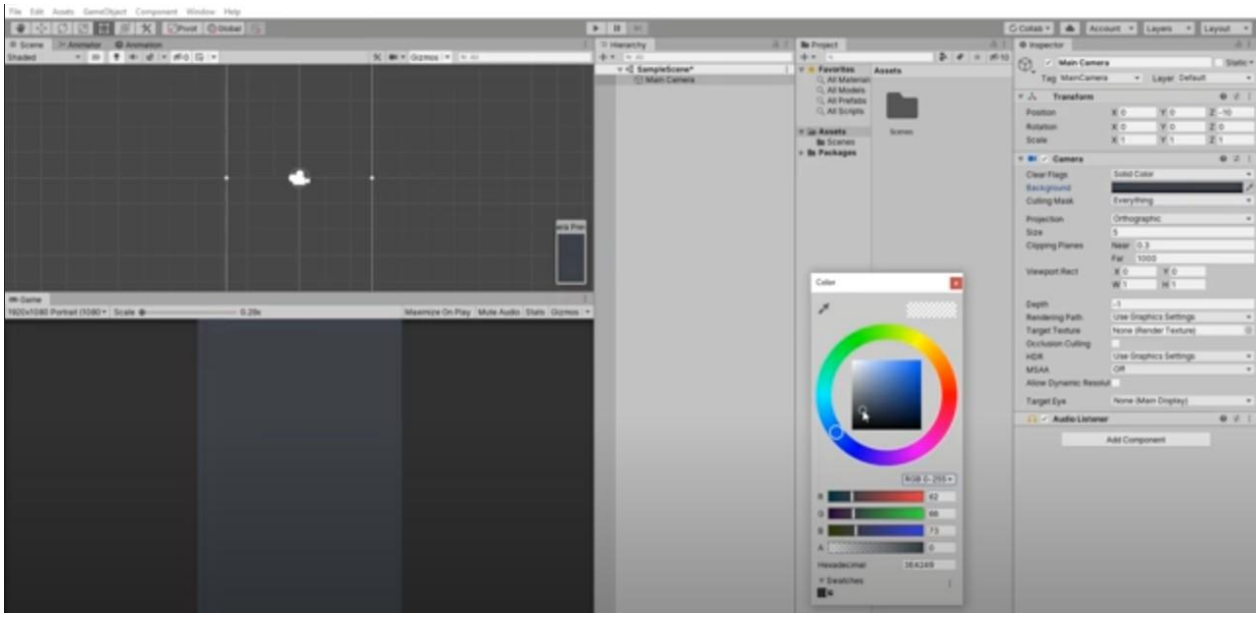

File
Edit
Assets
GameObject
Component
Window
Help
New Scene Ctrl+N
Open Scene Ctrl+O
Save Ctrl+S
Save As... Ctrl+Shift+S
New Project...
Open Project...
Save Project
Build Settings... Ctrl+Shift+B
Build And Run Ctrl+B
Exit
Pivot
Global
Gizmos
Game
Display 1
1080x1920
Scale
Maximize On Play
Mute Audio
Stats
Gizmos


SampleScene
Main Camera
Favorites
Assets
All Materials
Build Settings
Scenes In Build
Scenes/SampleScene
0
Add Open Scenes
Platform
PC, Mac & Linux Standalone
Android
Universal Windows Platform
tvOS
PS4
iOS
Xbox One
WebGL
Texture Compression
Don't override
ETC2 fallback
32-bit
Export Project
Build App Bundle (Google Play)
Development Build
Script Debugging
Patch
Patch And Run
Compression Method
LZ4
Learn about Unity Cloud Build
Player Settings...
Switch Platform
Build And Run


Build Settings
Scenes In Build
Scenes/SampleScene
Add Open Scenes
Platform
PC, Mac & Linux Standalone
Android
Universal Windows Platform
tvOS
PS4
iOS
Xbox One
WebGL
Texture Compression
ETC2 fallback
Export Project
Build App Bundle (Google Play)
Run Device
Development Build
Compression Method
Learn about Unity Cloud Build
Player Settings...
Build
Build And Run


Game
Scale
VSync (Game view only)
Free Aspect


Game
Scale
VSync (Game view only)
Free Aspect



Create a script called `PlayerController.cs` and attach it to the player character GameObject.

```
 using UnityEngine;

public class PlayerController : MonoBehaviour
{
    public float moveSpeed = 5f;
    private Rigidbody2D rb;

    void Start()
    {
        rb = GetComponent<Rigidbody2D>();
    }

    void Update()
    {
        float horizontalInput = Input.GetAxis("Horizontal");

        // Move the player horizontally
```

```
        rb.velocity = new Vector2(horizontalInput * moveSpeed, rb.velocity.y);
    }
}
```

Set Up the Platform:

Create a platform GameObject using GameObject > Create Empty. Rename it to "Platform" and attach a Box Collider 2D component to it.

Create Ground Check:

Attach an empty GameObject as a child of the player character. Rename it to "GroundCheck" and place it just below the player's feet. Attach a Circle Collider 2D to itand set it to be a trigger.

Modify the PlayerController Script:

Update the `PlayerController.cs` script to handle jumping and checking for ground collision.

```
using UnityEngine;

public class PlayerController : MonoBehaviour
{
    public float moveSpeed = 5f;
    public float jumpForce = 10f;
    private Rigidbody2D rb;
    private bool isGrounded = false;

    public Transform groundCheck;
    public LayerMask groundLayer;

    void Start()
    {
        rb = GetComponent<Rigidbody2D>();
    }

    void Update()
    {
```

```
        float horizontalInput = Input.GetAxis("Horizontal");
        isGrounded = Physics2D.OverlapCircle(groundCheck.position, 0.1f, groundLayer);

        / Move the player horizontally
        rb.velocity = new Vector2(horizontalInput * moveSpeed, rb.velocity.y);

        / Jump
        if (isGrounded && Input.GetButtonDown("Jump"))
        {
            rb.velocity = new Vector2(rb.velocity.x, jumpForce);
        }
    }
}
```

Set Up Input:

Go to Edit > Project Settings > Input Manager. Add a new Input Axis for "Horizontal" and "Jump."

Attach Components:

Attach the PlayerController script to the player character GameObject.
Assign the GroundCheck transform and set the Ground Layer in the inspector.

https://www.youtube.com/watch?v=t-rdSI3-Hfk
https://www.youtube.com/watch?v=4BD3y0NYNqk