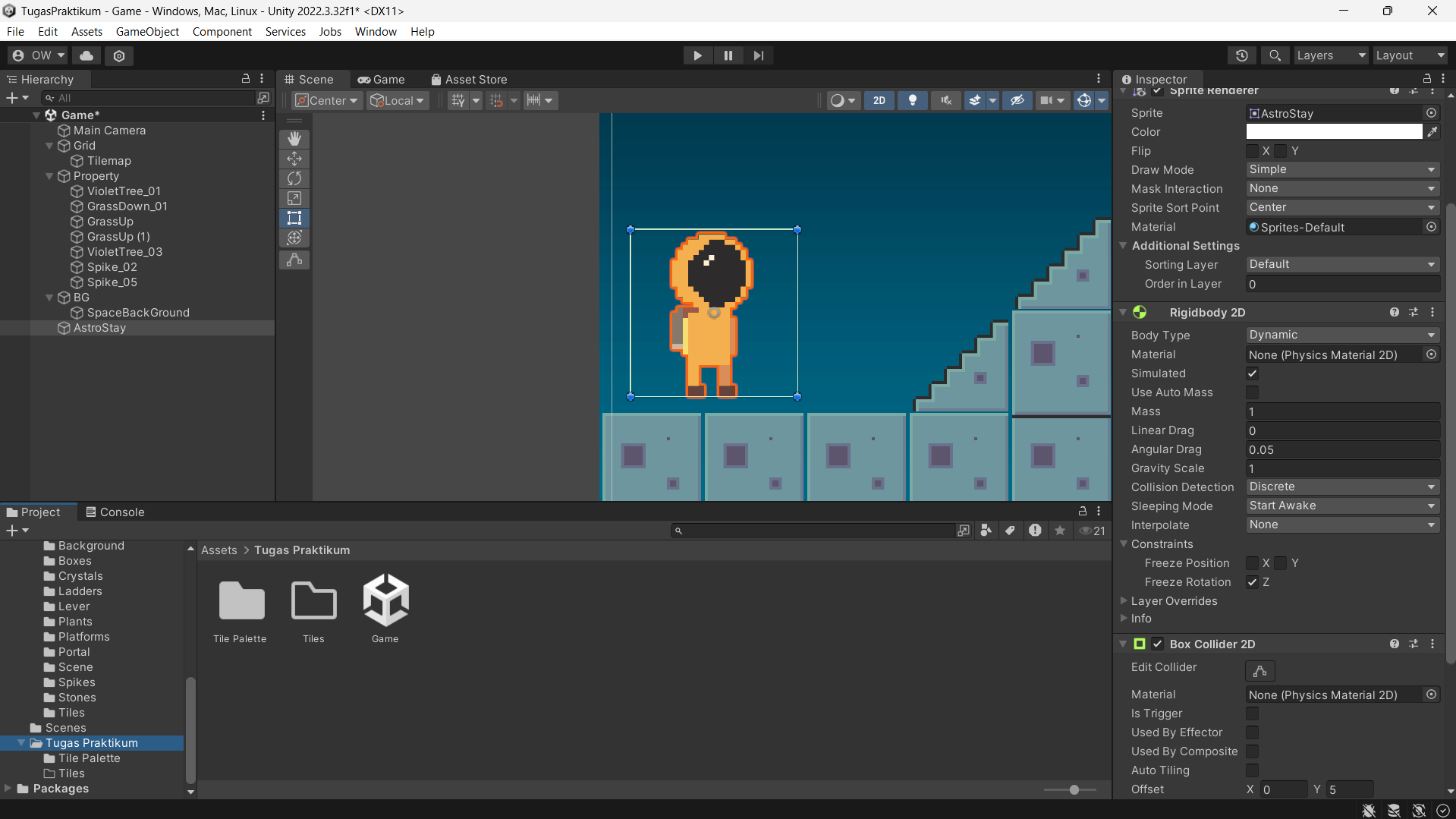
# 8 Camera & Character Movement

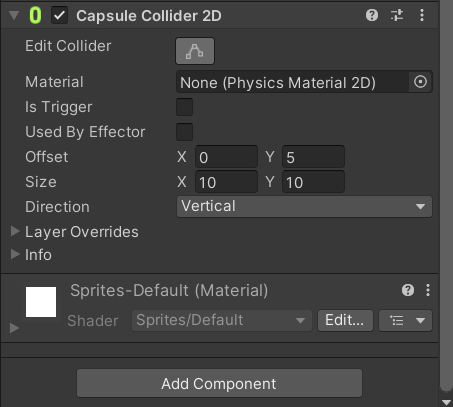
|  |  |  |
| --- | --- | --- |
| **NIM** | : | 2118017 |
| **Nama** | : | Jordano Saxonio |
| **Kelas** | : | A |
| **Asisten Lab** | : | Natasya octavia 2118034 |

## Tugas 1 : Camera & Character Movement

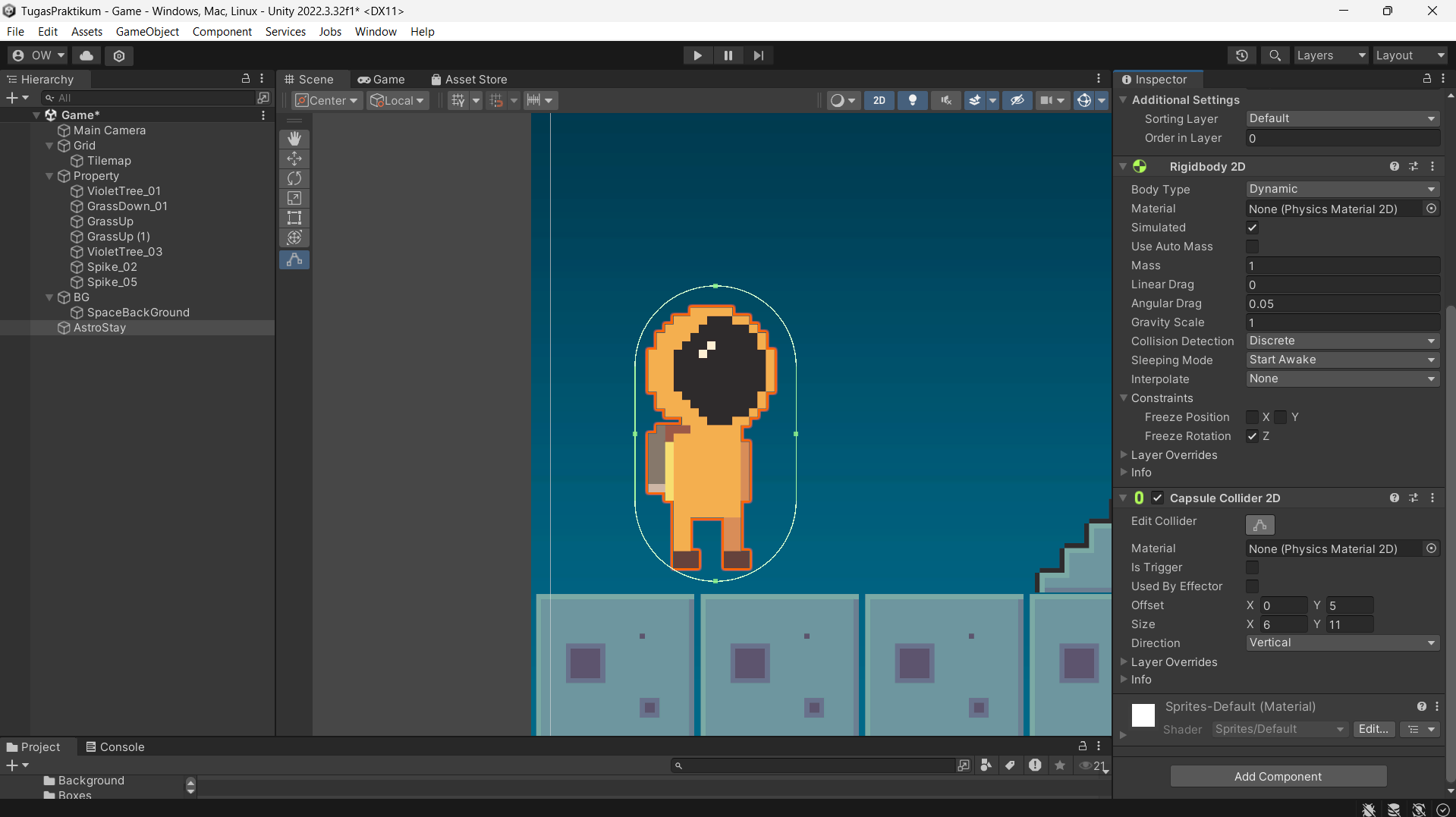
1. **Membuat Pergerakan Layer**
2. Klik AstroStay tambahkan component Rigidbody 2D, sesuaikan settingannya seperti gambar berikut, Centang pada Freeze Rotation Z.



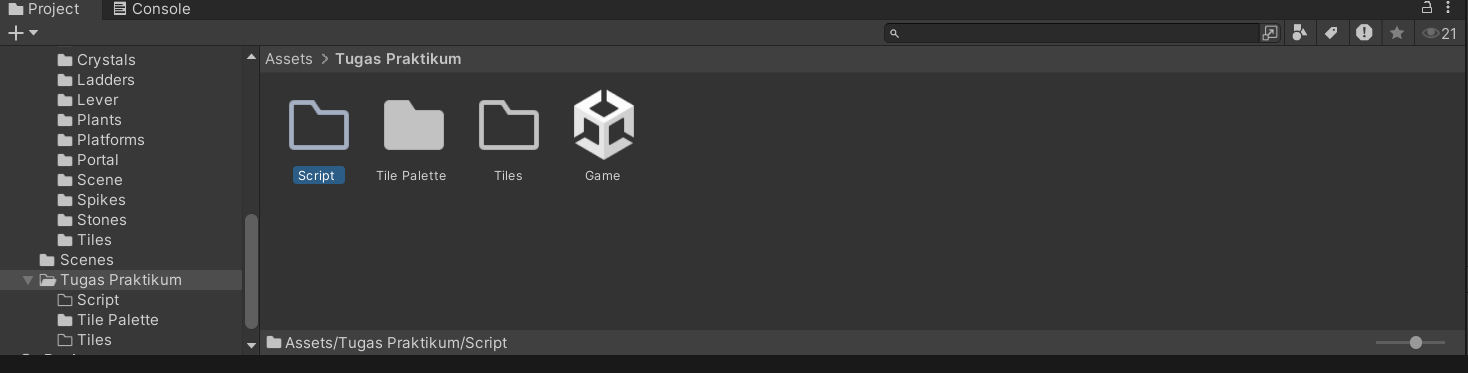
1. Lalu tambahkan komponen Capsule Colider di AstroStay, lalu klik icon sebelah kanan edit collider.



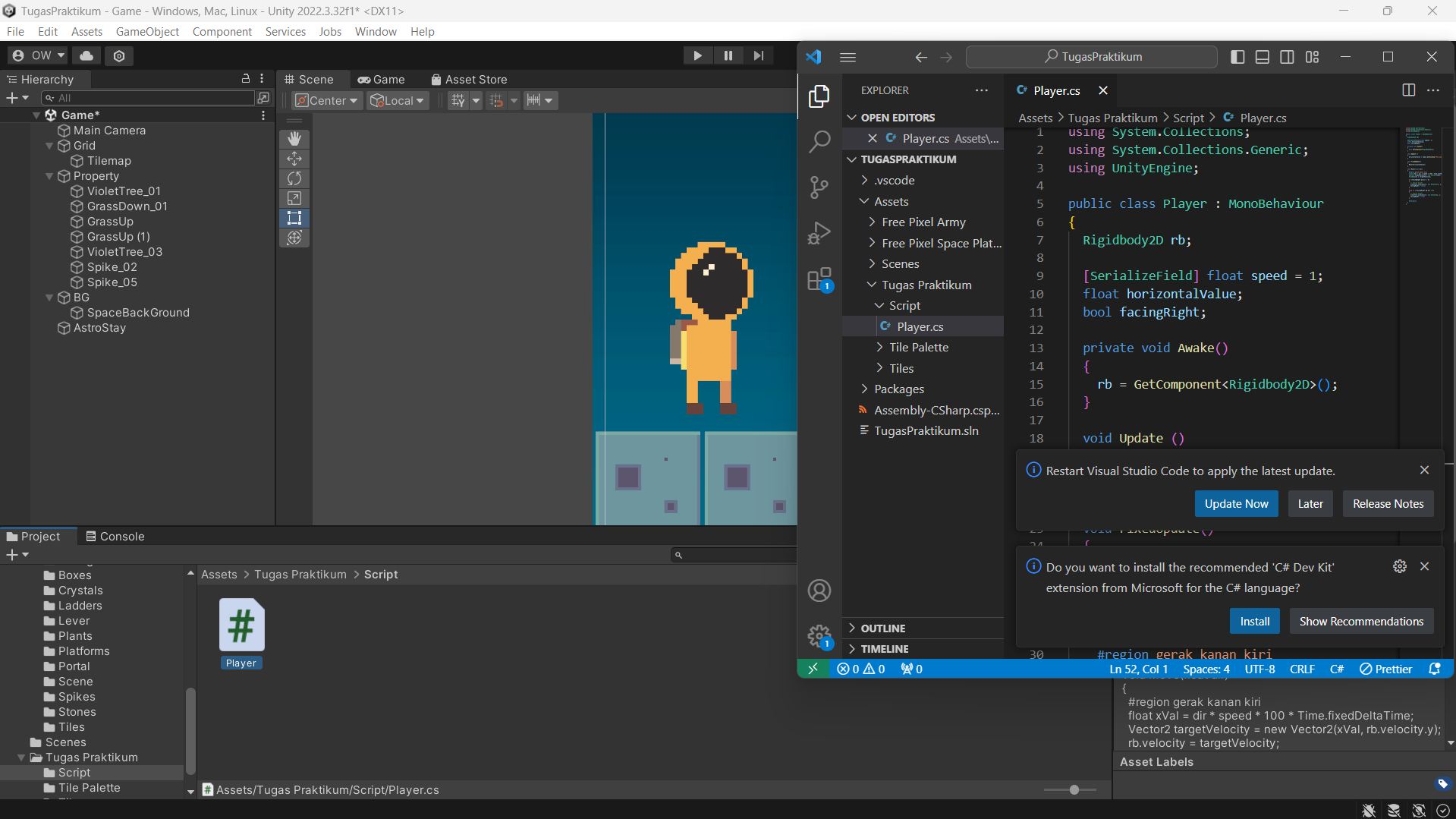
1. Lalu cocokan garis oval dengan karakternya atau bisa diinputkan Offset X, Y dan juga Size X, Y nya.



1. Buka folder TugasPraktikum, lalu bikin folder baru Bernama Script.



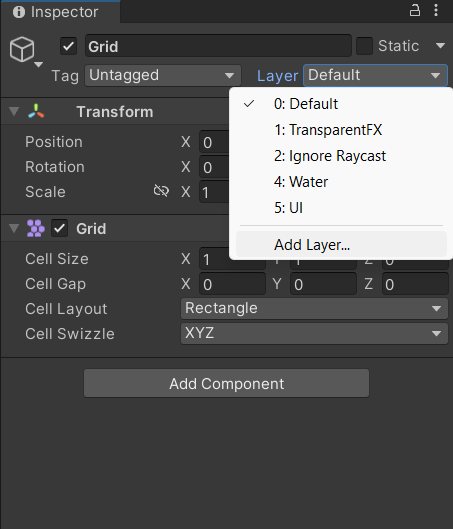
1. Masuk kedalam folder Script, lalu buat C# Script, beri nama Player. Drag & drop Script Player kedalam Hierarchy AstroWalk\_0, lalu klik 2x pada script Player maka akan masuk kedalam text editor seperti ini.

****

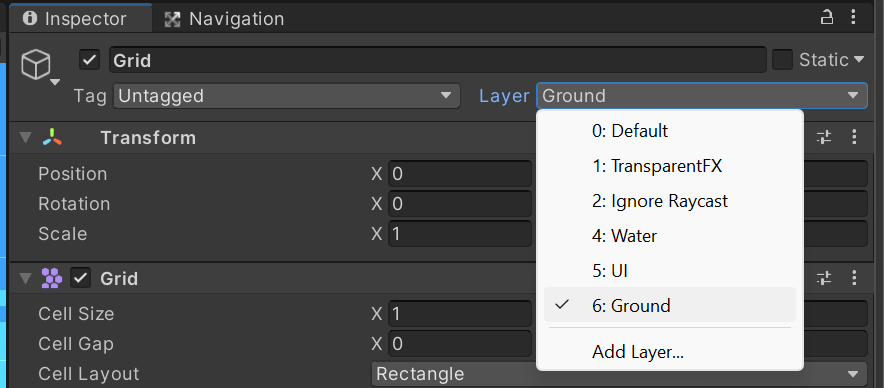
1. Masukkan source code dibawah ini, pastikan nama public class harus sama dengan nama file yang dibuat.

|  |
| --- |
| using System.Collections;  using System.Collections.Generic;  using UnityEngine;  public class Player : MonoBehaviour  {  Rigidbody2D rb;  [SerializeField] float speed = 1;  float horizontalValue;  bool facingRight;  private void Awake()  {  rb = GetComponent<Rigidbody2D>();  }  void Update ()  {  horizontalValue = Input.GetAxisRaw("Horizontal");  }  void FixedUpdate()  {  Move(horizontalValue);  }  void Move(float dir)  {  #region gerak kanan kiri  float xVal = dir \* speed \* 100 \* Time.fixedDeltaTime;  Vector2 targetVelocity = new Vector2(xVal, rb.velocity.y);  rb.velocity = targetVelocity;  if (facingRight && dir < 0)  {  // ukuran player  transform.localScale = new Vector3(-1, 1, 1);  facingRight = false;  }  else if (!facingRight && dir > 0)  {  // ukuran player  transform.localScale = new Vector3(1, 1, 1);  facingRight = true;  }  #endregion  }  } |

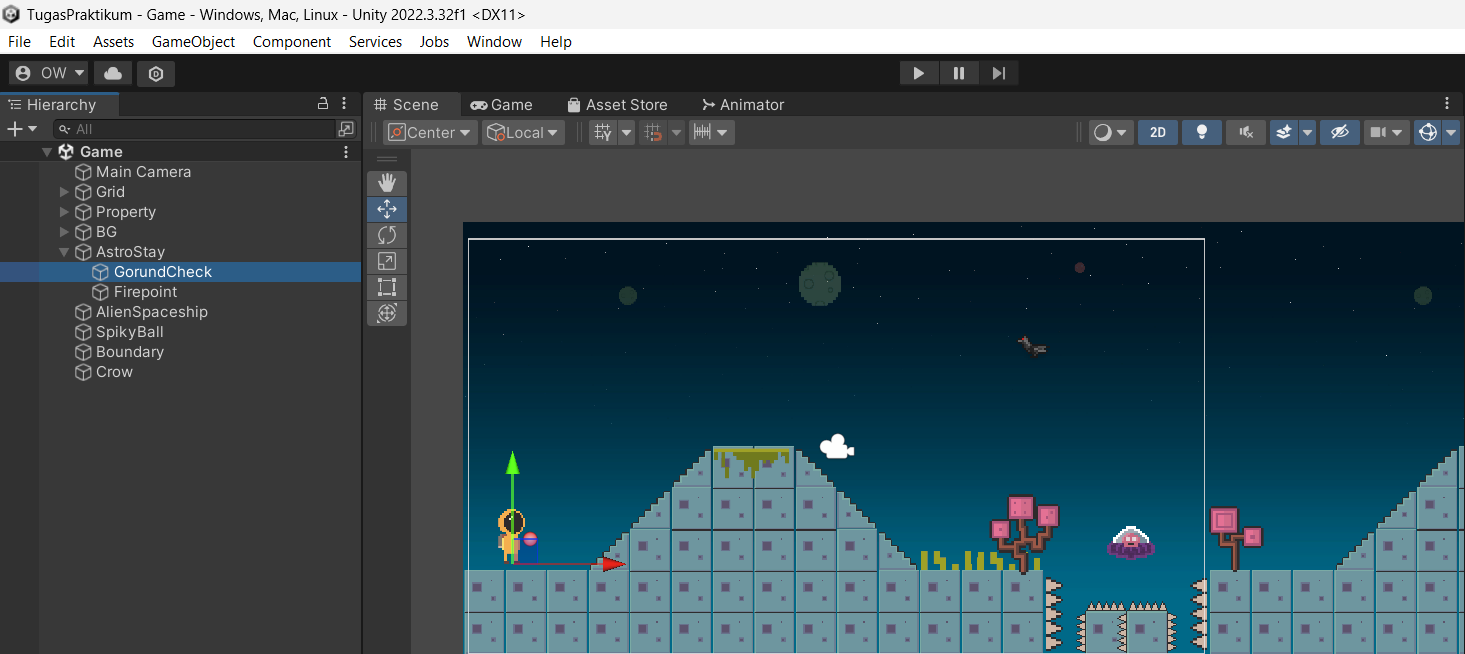
1. Klik Grid pada Hierarchy. Pergi ke inspector, pilih Layer, klik Add Layer.

****

1. Lalu isi “Ground” pada User Layer 6. Ubah Layer menjadi Ground, jika muncul pop up Change Layer, klik yes saja.

****

1. Klik kanan pada AstroStay, Pilih Create Empty. Klik pada Hierarchy GroundCheck, lalu gunakan “Move Tools” untuk memindahkan kebagian bawah Player seperti gambar berikut.

****

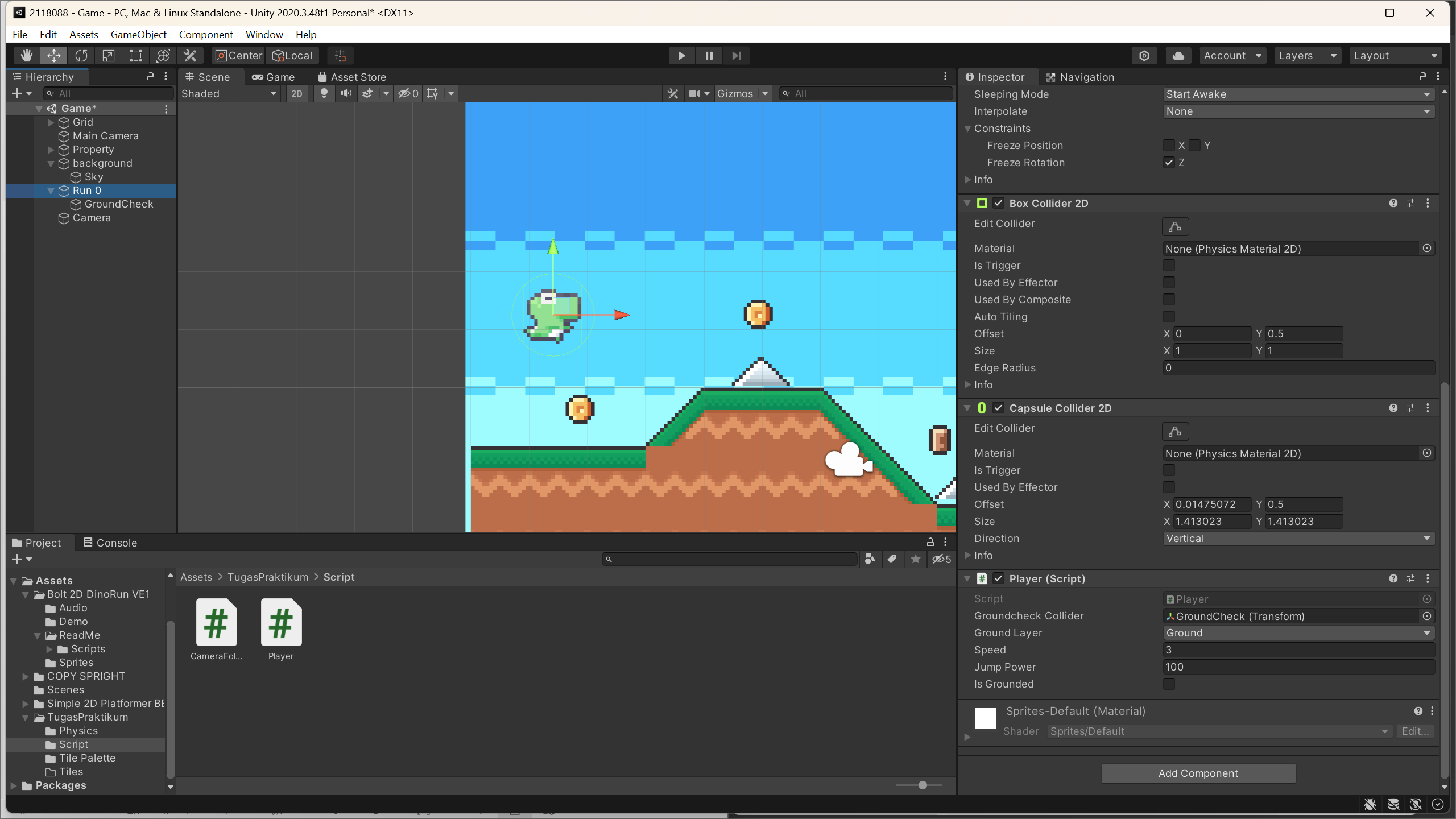
1. Kembali ke Script Player tambahkan source code seperti ini.

|  |
| --- |
| [SerializeField] Transform groundcheckCollider;  [SerializeField] LayerMask groundLayer;  const float groundCheckRadius = 0.2f; // +  [SerializeField] float speed = 1;  float horizontalValue;  [SerializeField] bool isGrounded; // +  bool facingRight; |

1. Buat Void GroundCheck dibawah coid fixedUpdate & tambahkan GroundCheck(); pada void FixedUpdate.

|  |
| --- |
| void FixedUpdate()  {  GroundCheck();  Move(horizontalValue);  }  void GroundCheck()  {  isGrounded = false;  Collider2D[] colliders = Physics2D.OverlapCircleAll(groundcheckCollider.position, groundCheckRadius, groundLayer);  if (colliders.Length > 0)  isGrounded = true;  } |

1. Klik AstroStay, lalu ke inspector ke effect Player script dibagian “Ground collider” tekan icon lalu pilih yang GroundCheck Transform, dan pada Ground Layer pilih Ground.

****

1. Lalu untuk membuat player melompat tambahkan script berikut.

|  |
| --- |
| [SerializeField] float jumpPower = 100;  bool jump; |

1. Tambahkan juga script berikut dibagian void update.

|  |
| --- |
| if (Input.GetButtonDown("Jump"))  jump = true;  else if (Input.GetButtonUp("Jump"))  jump = false; |

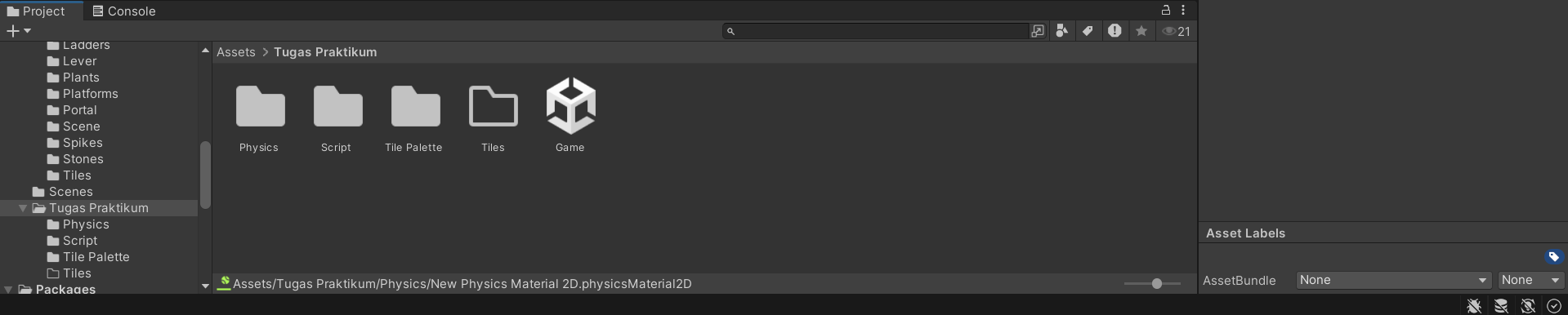
1. Tambahkan juga jump pada parameter Move.

|  |
| --- |
| void FixedUpdate(){  GroundCheck();  Move(horizontalValue, jump);  } |

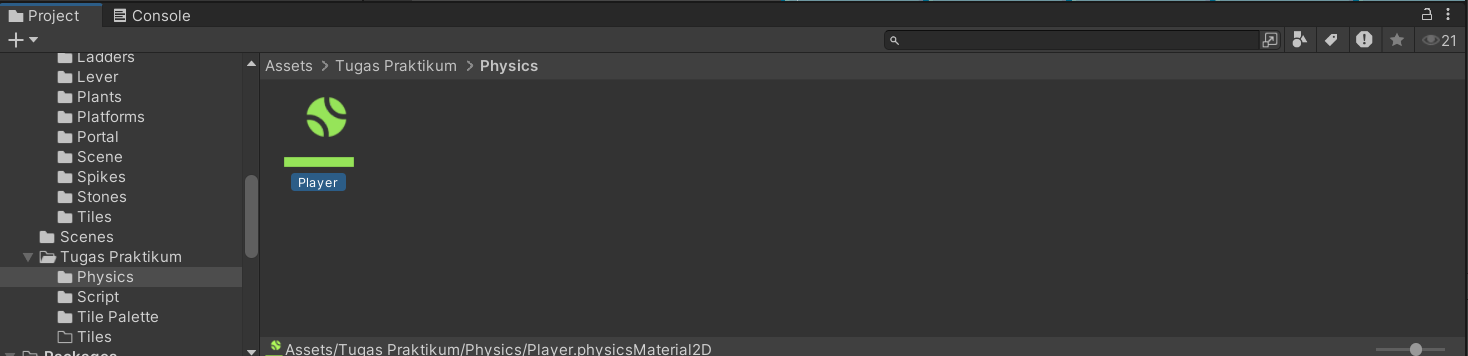
1. Tambahkan script berikut pada void Move.

|  |
| --- |
| bool jumpflag  if(isGrounded && jumpflag)  {  isGrounded = false;  jumpflag = false;  rb.AddForce(new Vector2(0f, jumpPower));  } |

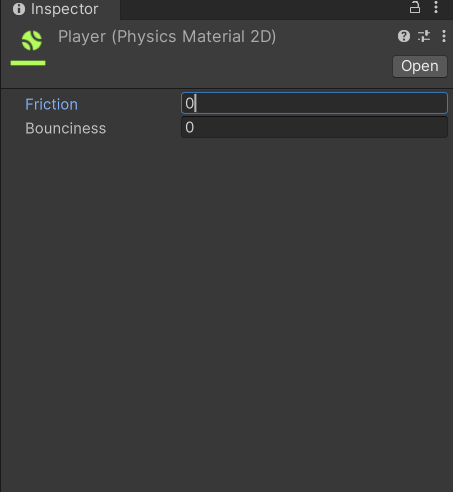
1. Buat folder baru di TugasPraktikum Bernama “Physics”.

****

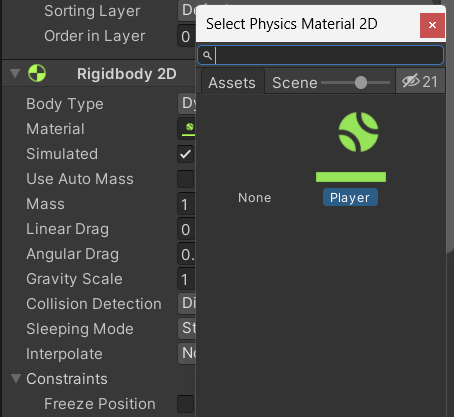
1. Didalam folder Physhics create >2D > Physhical Material 2D, berinama “Player”.

****

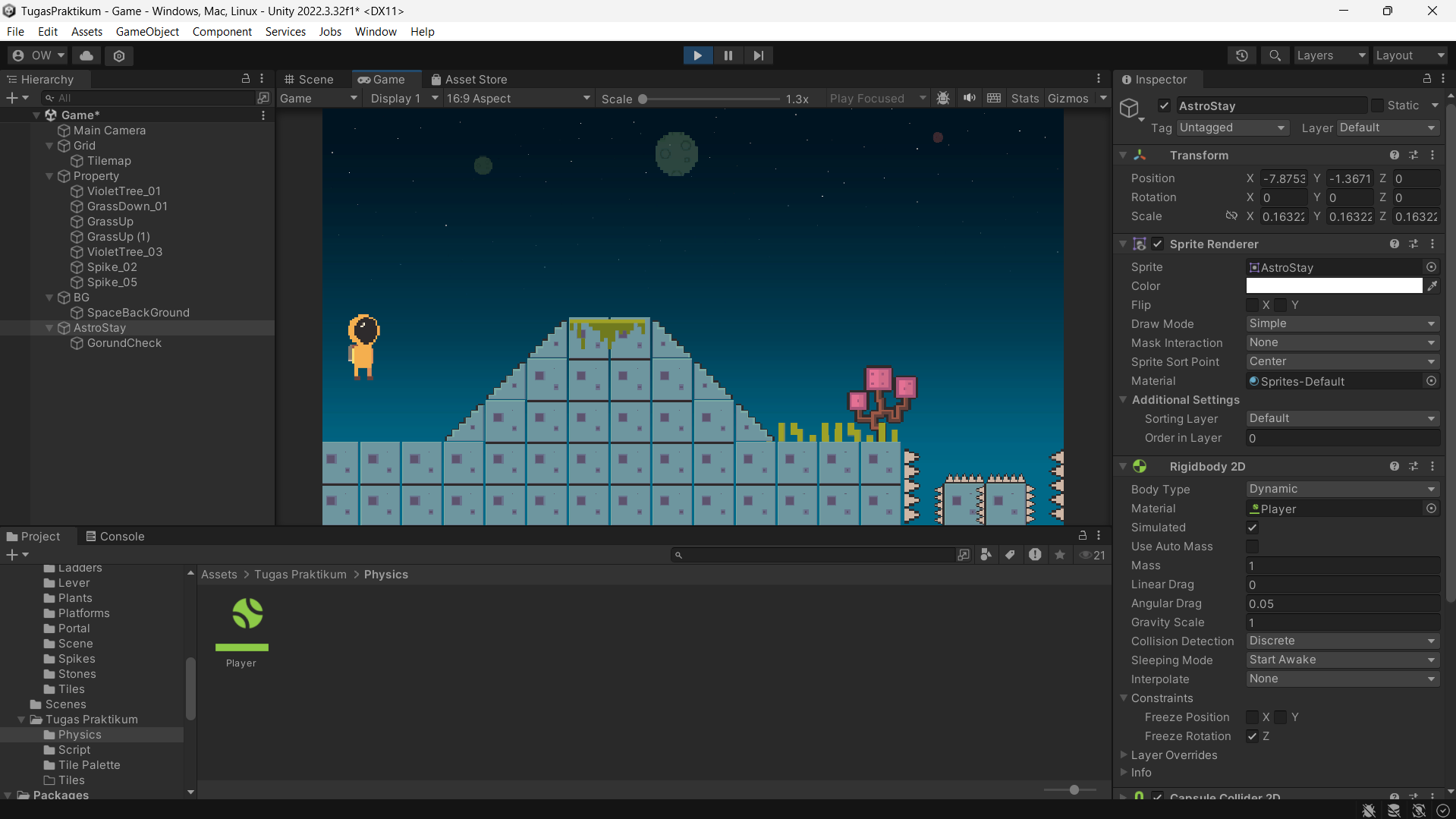
1. Klik Player (Physhics Material 2D). dibagian menu inspector, friction & bounces ubah menjadi 0.

****

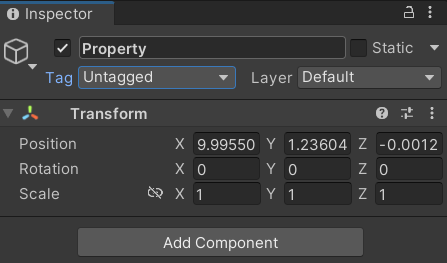
1. Klik Hierarchy pilih layer Run 0, pada inspector cari RigidBody2D lalu klik icon unttuk membuka box select Physhics Material 2D, lalu pilih asset Player yang sudah dibuat tadi.

****

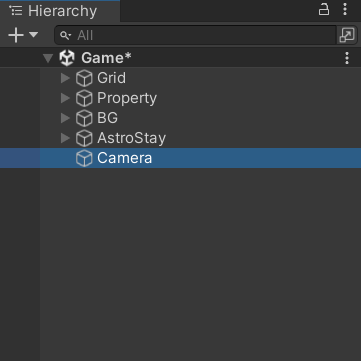
1. Tekan play, maka Player bisa melompat dengan menekan spasi.

****

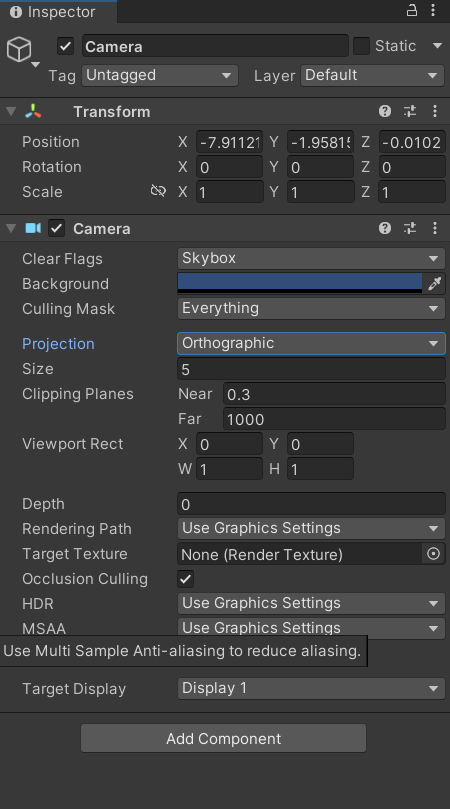
1. **Camera Movement**
2. Pada Hierarchy Property ubah inspector pada tag Main camera menjadi untagged.

****

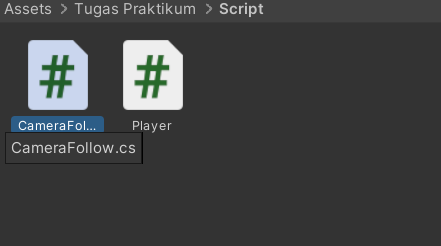
1. Create Empty pada Hierarchy, dan renam menjadi Camera.

****

1. Sesuaikan Setting Layer Camera seperti gambar dibawah ini.

****

1. Buat file script baru difolder script dengan nama “CameraFollow”.

****

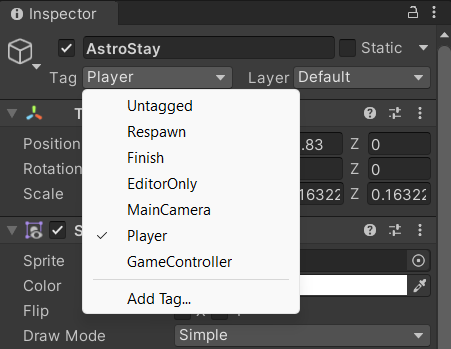
1. Lalu tuliskan script berikut ini.

|  |
| --- |
| using System.Collections;  using System.Collections.Generic;  using UnityEngine;  public class CameraFollow : MonoBehaviour  {  public float xMargin = 0.5f;  public float yMargin = 0.5f;  public float xSmooth = 4f;  public float ySmooth = 4f;  public Vector2 maxXAndY;  public Vector2 minXAndY;  private Transform player;  void Awake()  {  player = GameObject.FindGameObjectWithTag("Player").transform;  }  bool CheckXMargin()  {  return Mathf.Abs(transform.position.x - player.position.x) > xMargin;  }  bool CheckYMargin()  {  return Mathf.Abs(transform.position.y - player.position.y) > yMargin;  }  void FixedUpdate()  {  TrackPlayer();  }  void TrackPlayer()  {  float targetX = transform.position.x;  float targetY = transform.position.y;  if (CheckXMargin())  targetX = Mathf.Lerp(transform.position.x, player.position.x,  xSmooth \* Time.deltaTime);  if (CheckYMargin())  targetY = Mathf.Lerp(transform.position.y, player.position.y,  ySmooth \* Time.deltaTime);  targetX = Mathf.Clamp(targetX, minXAndY.x, maxXAndY.x); targetY =  Mathf.Clamp(targetY, minXAndY.y, maxXAndY.y); transform.position = new  Vector3(targetX, targetY, transform.position.z);  }  } |

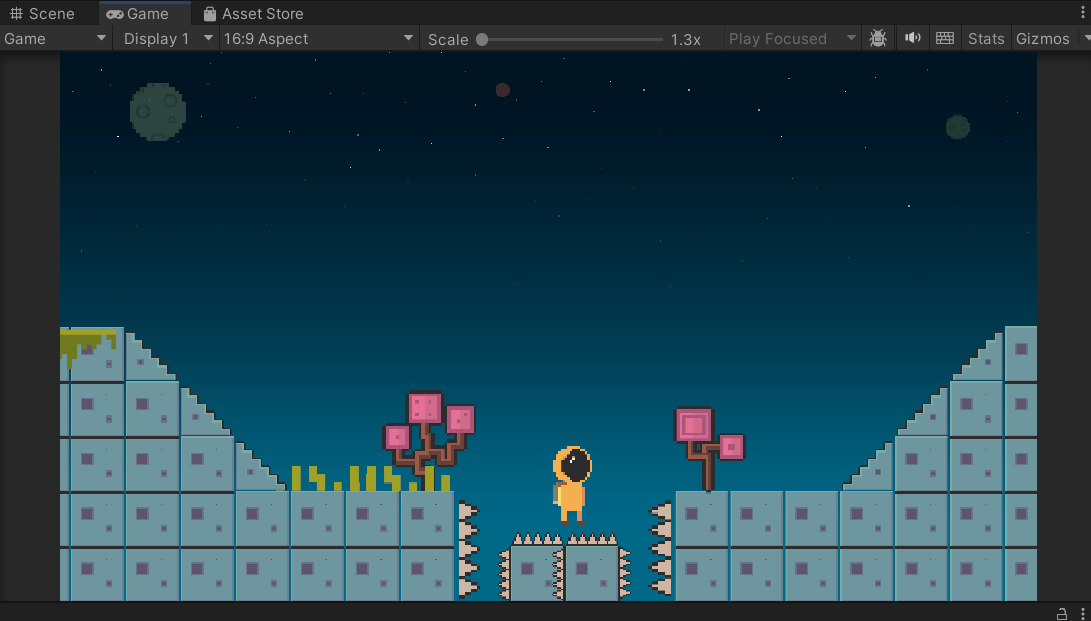
1. Drag & drop script CameraFollow kedalam Layer Camera. Lalu kill pada Camera, buka inspector pada bagian CameraFollow (Script) ubah Bagian Max X dan Max Y nya.



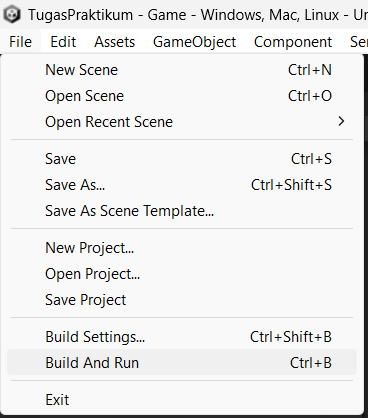
1. Ubah tag di Run 0 Untagged menjadi “Player”.



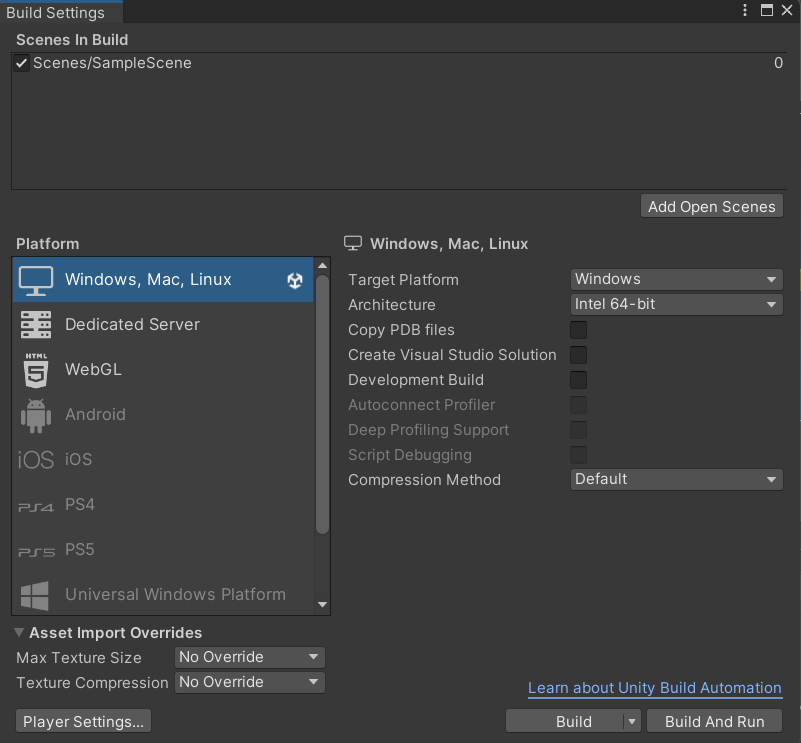
1. Tekan play untuk menjalankan, maka sekarang kamera akan mengikuti pergerakan karakter.



1. **Render**
2. Pergi ke menu File kemudioan pilih Build Setting.

****

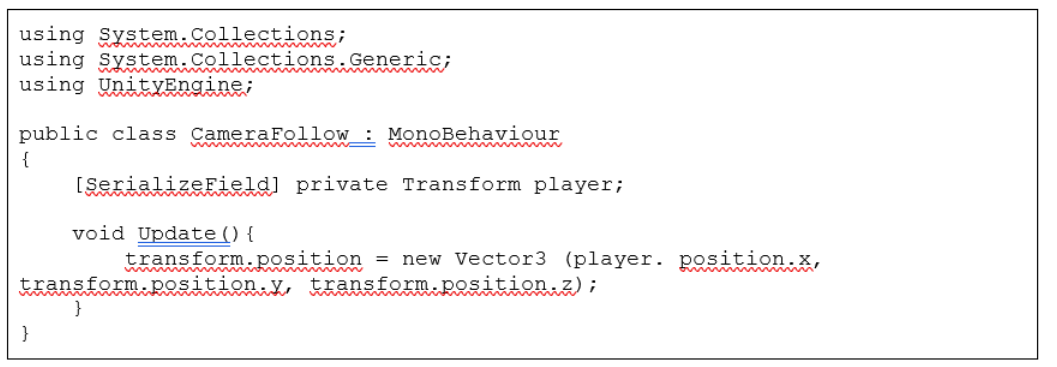
1. Pada setting Build ini pilih PC, Mac & Linux, Tekan Build, pastikan pada menu Scene in Build, pasttikan pada menu Scene ion Build berada pada project Tugas.

****

1. Hasil setelah dirender.

****

1. **Kuis**

****

Codingan diatas digunakan untuk membuat kamera mengikuti pergerakan dari pemain secara horizontal (sumbu x), SerializeField sendiri memungkinkan kita mengubah referensi dari “player” langsung dari inspector yang ada pada unity. void Update() sendiri dipanggil sekali pada setiap frame didalam unity dan posisi kamera akan diperbarui agar selalu memiliki nilai x yang sama dengan posisi pemain, sementara untuk nilai y dan z akan tetap sama.