



TUGAS PERTEMUAN: 8

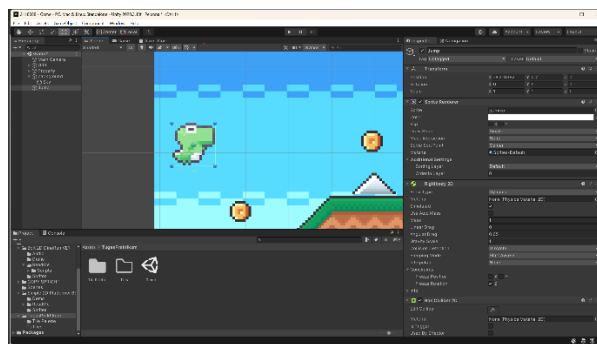
Camera & Character Movement

NIM	:	2118088
Nama	:	Nuangga Ervin Dwi Syahputra
Kelas	:	A
Asisten Lab	:	Nur Aria Hibnastiar (2118078)

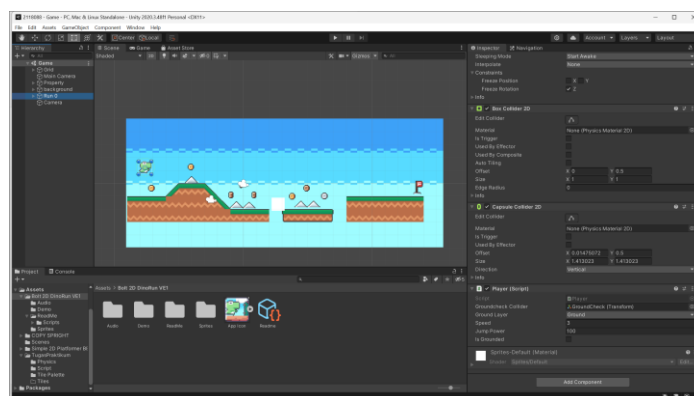
8.1 Tugas 1 : Camera & Character Movement

A. Membuat Pergerakan Layer

1. Klik Run 0 tambahkan component Rigidbody 2D, sesuaikan settingannya seperti gambar berikut, Centang pada Freeze Rotation Z.

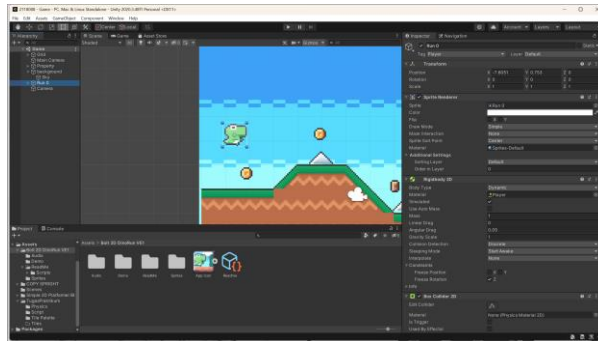


2. Lalu tambahkan komponen Capsule Colider di Run 0, lalu klik icon sebelah kanan edit collider.

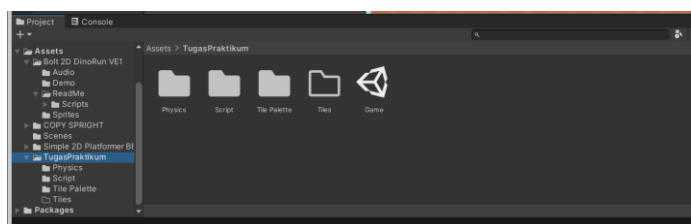




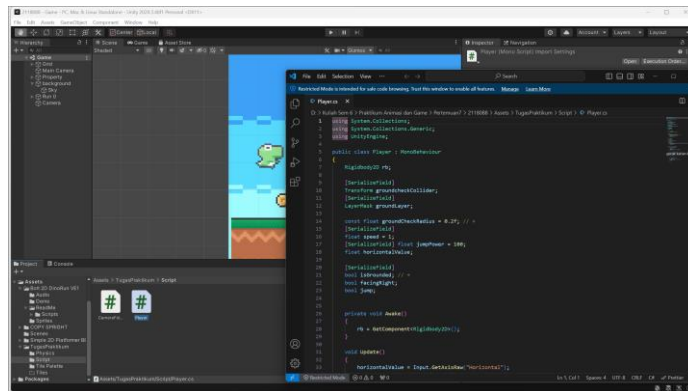
3. Lalu cocokkan garis oval dengan karakternya atau bisa diinputkan Offset X, Y dan juga Size X, Y nya.



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



5. Masuk kedalam folder Script, lalu buat C# Script, beri nama Player. Drag & drop Script Player kedalam Hierarchy Run 0, lalu klik 2x pada script Player maka akan masuk kedalam text editor seperti ini.



6. 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);
}

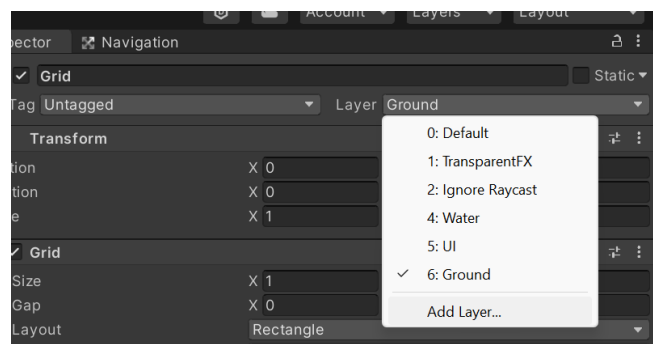
void Move(float dir)
{
    #region gerak kanan kiri
    float xVal = dir * speed * 100 * Time.fixedDeltaTime;
    Vector2 targetVelocity = new Vector2(xVal, rb.velocity.y);
    rb.velocity = targetVelocity;

    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;
    }

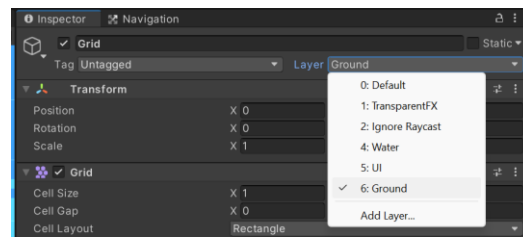
    #endregion
}
}
```

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

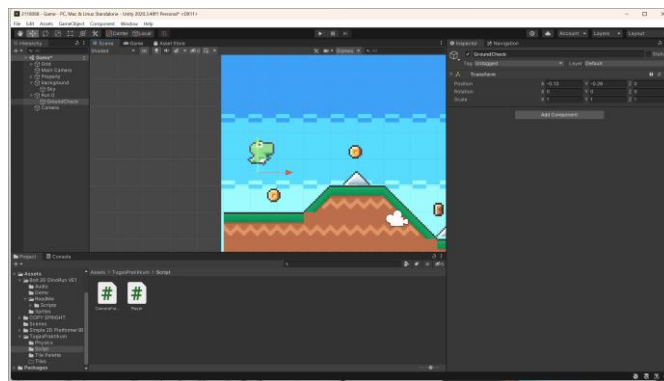




8. Lalu isi “Ground” pada User Layer 6. Ubah Layer menjadi Ground, jika muncul pop up Change Layer, klik yes saja.



9. Klik kanan pada Run 0, Pilih Create Empty. Klik pada Hierarchy GroundCheck, lalu gunakan “Move Tools” untuk memindahkan sebagian bawah Player seperti gambar berikut.



10. 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;
```

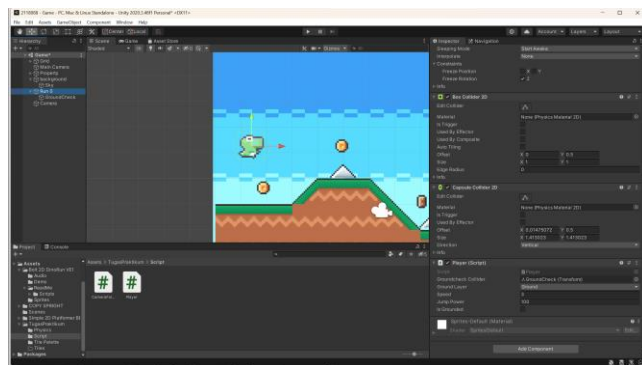


11. Buat Void GroundCheck dibawah coid fixedUpdate & tambahkan GroundCheck(); pada void FixedUpdate.

```
void FixedUpdate()
{
    GroundCheck();
    Move(horizontalValue);
}

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

12. Klik Run0, lalu ke inspector ke effect Player script dibagian “Ground collider” tekan icon lalu pilih yang GroundCheck Transform, dan pada Ground Layer pilih Ground.



13. Lalu untuk membuat player melompat tambahkan script berikut.

```
[SerializeField] float jumpPower = 100;

bool jump;
```

14. Tambahkan juga script berikut dibagian void update.

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

15. Tambahkan juga jump pada parameter Move.

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

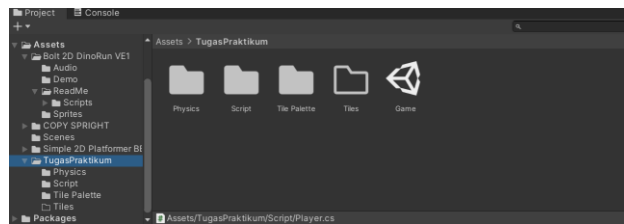


16. Tambahkan script berikut pada void Move.

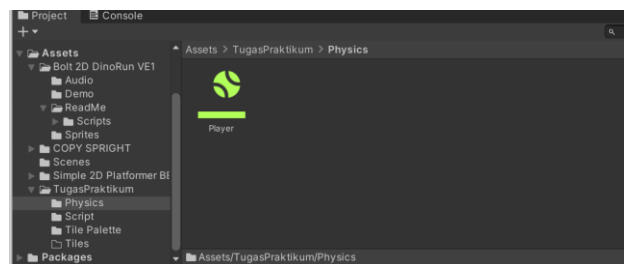
```
bool jumpflag

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

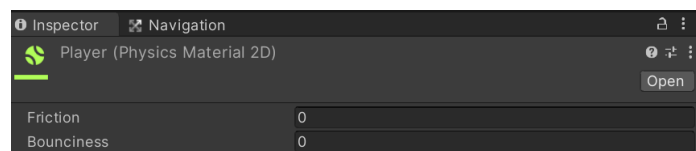
17. Buat folder baru di TugasPraktikum Bernama “Physics”.



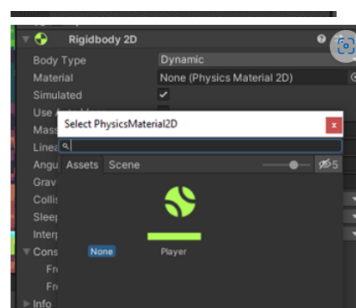
18. Didalam folder Physics create >2D > Physshical Material 2D, bernama “Player”.



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

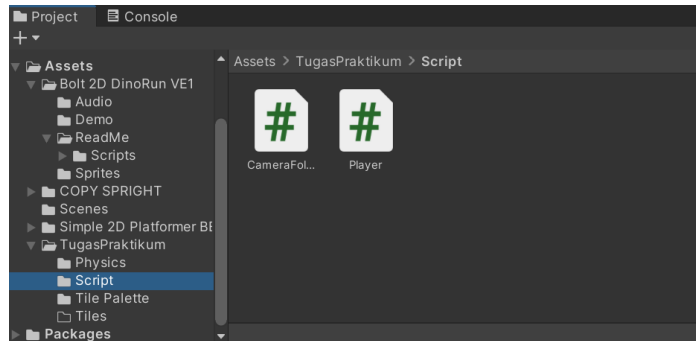


20. Klik Hierarchy pilih layer Run 0, pada inspector cari Rigidbody2D lalu klik icon untuk membuka box select Physshics Material 2D, lalu pilih asset Player yang sudah dibuat tadi.





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



5. 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;
    }

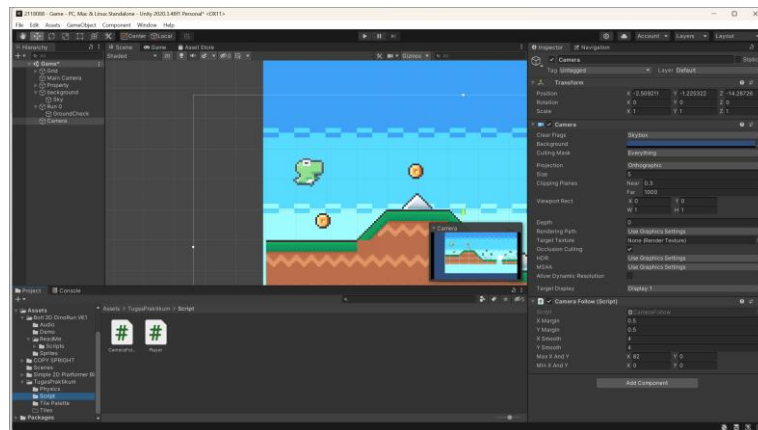
    void FixedUpdate()
    {
        TrackPlayer();
    }

    void TrackPlayer()
    {
        float targetX = transform.position.x;
        float targetY = transform.position.y;
        if (CheckXMargin())
```

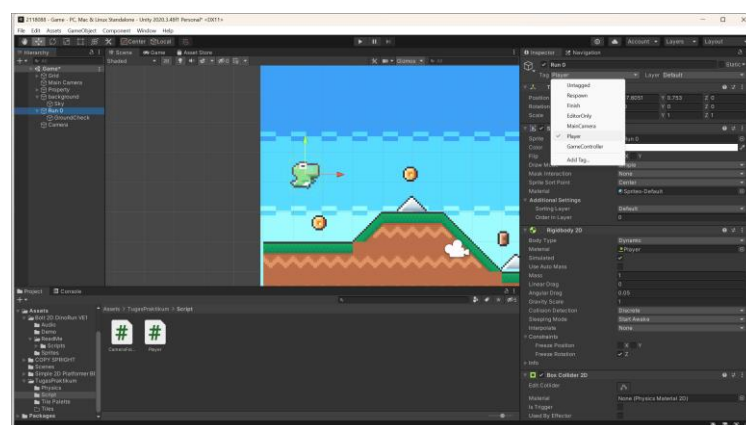



```
targetX = Mathf.Lerp(transform.position.x,
player.position.x,
xSmooth * Time.deltaTime);
if (CheckYMargin())
targetY = Mathf.Lerp(transform.position.y,
player.position.y,
ySmooth * Time.deltaTime);
targetX = Mathf.Clamp(targetX, minXAndY.x,
maxXAndY.x); targetY =
Mathf.Clamp(targetY,
minXAndY.y,
maxXAndY.y); transform.position = new
Vector3(targetX,
targetY,
transform.position.z);
}
```

6. 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.

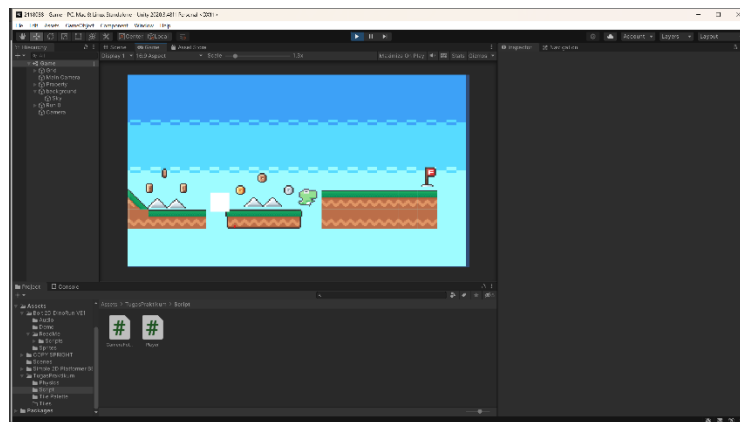


7. Ubah tag di Run 0 Untagged menjadi “Player”.



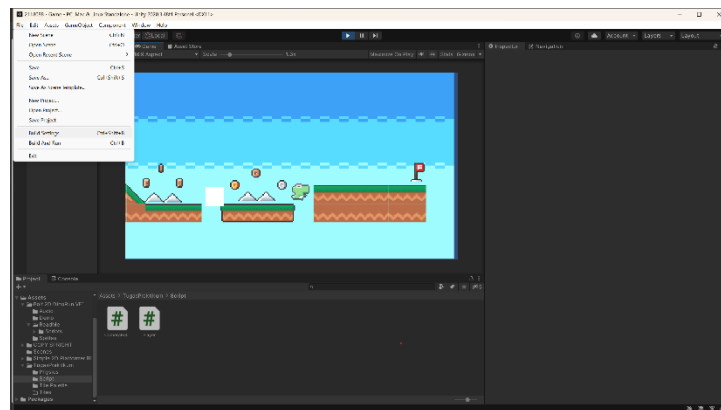


8. Tekan play untuk menjalankan, maka sekarang kamera akan mengikuti pergerakan karakter.

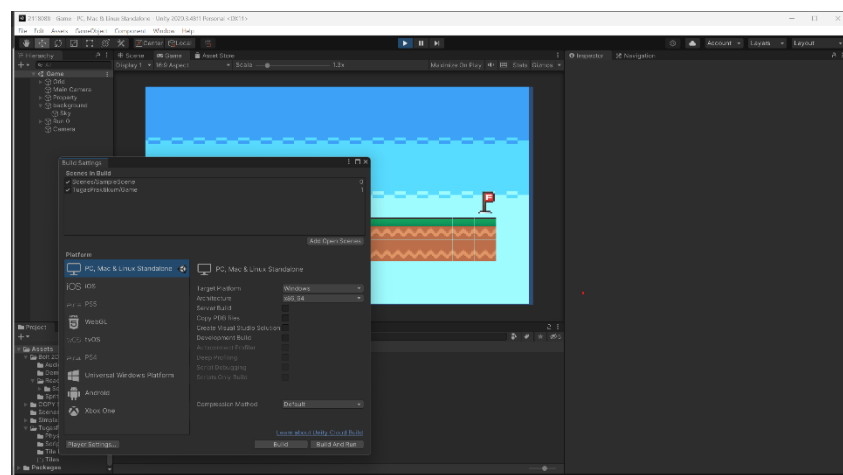


C. Render

1. Pergi ke menu File kemudian pilih Build Setting.

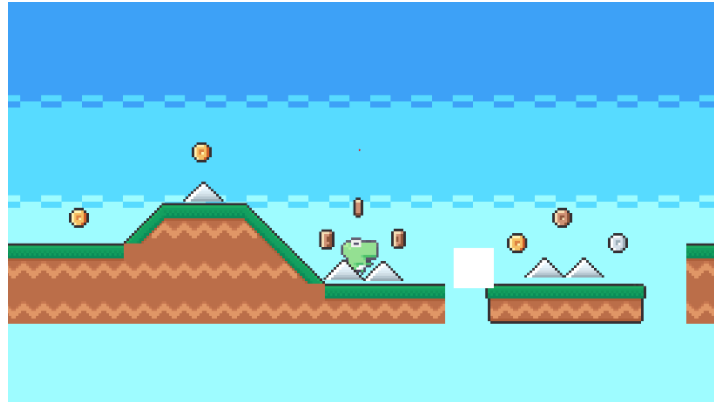


2. Pada setting Build ini pilih PC, Mac & Linux, Tekan Build, pastikan pada menu Scene in Build, pastikan pada menu Scene ion Build berada pada project Tugas.





3. Hasil setelah dirender.



D. Kuis

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

public class CameraFollow : MonoBehaviour
{
    [SerializeField] private Transform player;

    void Update() {
        transform.position = new Vector3 (player.position.x,
        transform.position.y, transform.position.z);
    }
}
```

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.

E. Link Github Pengumpulan

https://github.com/NuanggaErvinDwiSyahputra/2118088_PRAK_ANIGA_ME.git

F. Link Kuis

https://github.com/NuanggaErvinDwiSyahputra/2118088_PRAK_ANIGA_ME.git