



Điều khiển nhân vật bằng Keyboard



Một số hàm thường dùng

- Start()
- Awake()
- Update()
- FixedUpdate()
- OnCollisionEnter2D(Collision2D collision)

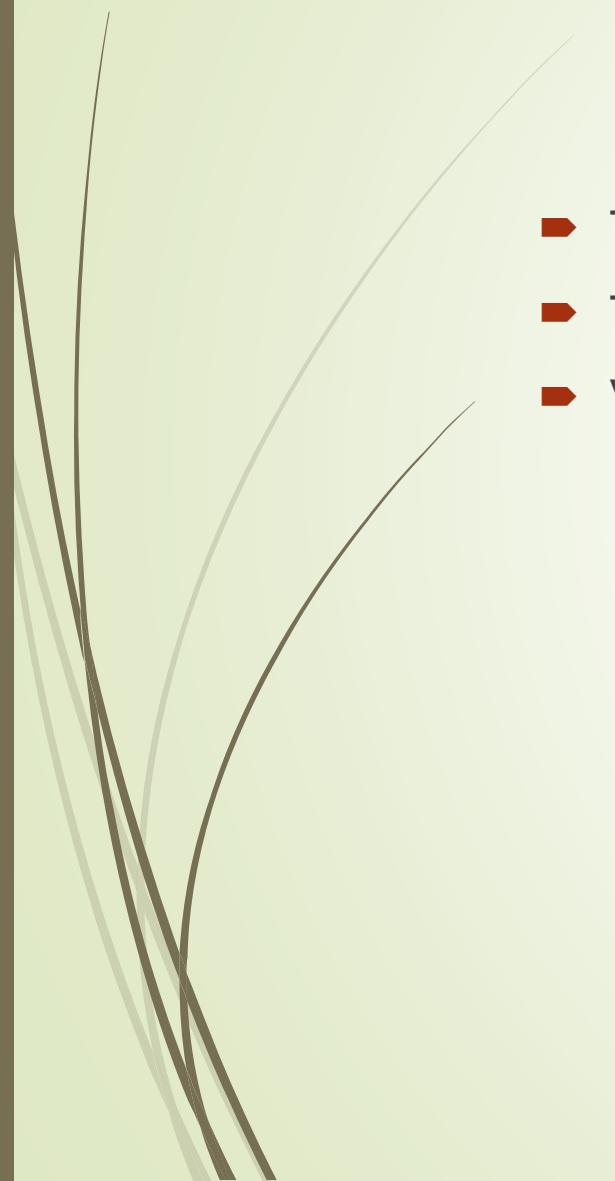


Một số hàm thường dùng

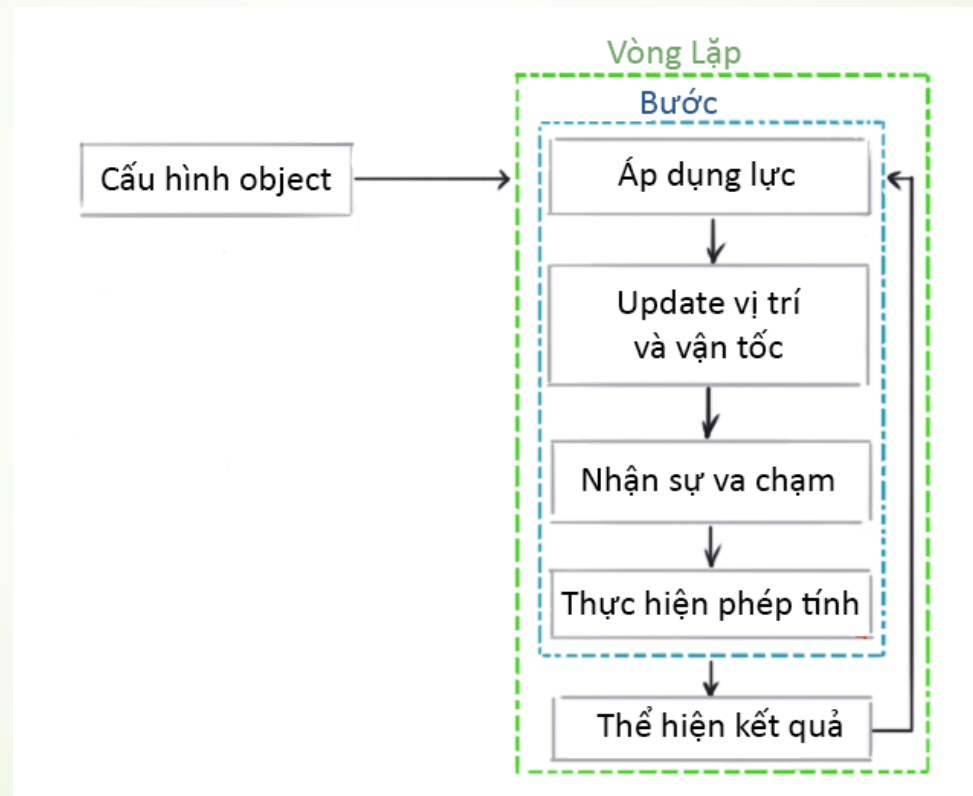
- `Input. GetMouseButtonDown()`
- `Input.GetKeyDown(keyCode), GetKey(keyCode), GetKeyUp(keyCode)`
- `Input.GetAxisRow()`
- `gameObject.transform(.position, .scale, .rotate)`
- `Destroy(GameObject)`
- `Instantiate(gameObject, Vector3, Quaternion)`



Điều khiển Player

- Thêm component Rigidbody2D
 - Thêm component BoxCollider2D
 - Viết script điều khiển
- 

Kiến trúc vật lý




Rigidbody: đặc tính vật lý

- Mass – Khối lượng của vật.
- **Linear Drag**: Hệ số ma sát của vật đối với chuyển động kéo, 0 nghĩa là hoàn toàn không có sức cản, vô tận sẽ khiến cho object ngừng di chuyển.
- **Angular Drag**: Hệ số ma sát của vật đối với chuyển động quay (ma sát lăn)
- Use Gravity – trọng lực sẽ được áp dụng lên object.
- Is Kinematic – loại bỏ tác dụng vật lý ra khỏi đối tượng, thường sử dụng với các đối tượng như tường, nền ...
- Interpolate – Dùng để điều chỉnh sự va chạm, làm smooth sự chuyển động
- Collision Detection – Dùng để ngăn chặn các object di chuyển quá nhanh xuyên qua các object khác mà không bị va chạm.
- **Fixed Angle**: Đối tượng luôn nằm một góc cố định. Không thay đổi khi tương tác vật lý.
- **rigidbody2D.AddForce(Vector2 f)**: phương thức tác dụng một lực vào đối tượng



Tham chiếu đến Component

- `myBody = GetComponent<Rigidbody2D>();`
- `anim = GetComponent<Animator>();`



Collider: xử lý va chạm

- Circle, Box, Edge, Polygon Collider
- Is Trigger: nếu chọn thì đối tượng chỉ dùng để xác định va chạm mà không ảnh hưởng bởi tác động vật lý.
- Fixed Angle



Các bước thực hiện demo nhân vật di chuyển, nhảy.

- Tạo nhân vật
- Thiết kế các nền đất (ground, Platform) đặt tag là “ground”
- Thiết lập component BoxCollider2D cho Ground, Platform, Player
- Thiết lập component Rigidbody2D cho Player
 - Thiết lập các thuộc tính trọng lực, ma sát, Freez Rotation cho Player
- Viết script cho player



Class Player

```
public class PlayerScript : MonoBehaviour
{

    public float moveForce = 20f;
    public float maxVelocity = 4f;

    private Rigidbody2D myBody;
    private Animator anim;
```



Class Player

```
void Awake()  
{  
    myBody = GetComponent<Rigidbody2D>();  
    anim = GetComponent<Animator>();  
}
```

```
// Use this for initialization  
void Start()  
{  
  
}
```



Class Player

```
// Update is called once per frame  
void FixedUpdate()  
{  
    PlayerWalkKeyBoard ();  
}
```

Class Player

```
void PlayerWalkKeyBoard() {  
    float forceX = 0f;  
    float forceY = 0f;  
    float vel = Mathf.Abs(myBody.velocity.x);  
    float h = Input.GetAxisRaw("Horizontal");  
  
    if (h > 0)  
    {  
        if (vel < maxVelocity)  
        {  
            forceX = moveForce;  
        }  
    }  
}
```

```
    else if (h < 0)  
    {  
        if (vel < maxVelocity)  
        {  
            forceX = -moveForce;  
        }  
    }  
    myBody.AddForce(new Vector2(forceX,0));  
}
```



Điều khiển nhân vật nhảy lên

```
//Nếu nhân vật chạm đất thì tác động một lực theo chiều y
void FixedUpdate()
{
    If(Input.GetKey(KeyCode.Space))
    {
        if(grounded){
            grounded=false;
            forceY = jumoForce;
        }
        myBody.addForce(new Vector2(forceX, forceY));
    }
}
```



Điều khiển nhân vật nhảy lên

```
//Nếu nhân vật chạm đất thì gán biến grounded=true  
void OnCollisionEnter2D(Collision2D collision)  
{  
    if(collision.gameObject.tag == "Ground") grounded=true;  
}
```

Điều khiển nhân vật nhảy lên

```
void PlayerKeyboard()  
{  
    float h = Input.GetAxisRaw("Horizontal");  
    float vel = Mathf.Abs(body.velocity.x);  
    float x = 0, y = 0;  
    if (h > 0)  
    {  
        if (vel < 4f)  
            x = forceX;  
        Vector3 v = transform.localScale;  
        v.x = 1f;  
        transform.localScale = v;  
    } else
```

```
    if (h < 0) {  
        if (vel < 4f)  
            x = -forceX;  
        Vector3 v = transform.localScale;  
        v.x = -1f;  
        transform.localScale = v;  
    }  
    if (Input.GetKey(KeyCode.Space))  
        if (grounded)  
        {  
            y = forceY;  
            grounded = false;  
        }  
        body.AddForce(new Vector2(x, y)); // dùng addforce nếu muốn vật  
        có quán tính di chuyển tiếp,  
    }
```




Điều khiển nhân vật nhảy lên

```
body.AddForce(new Vector2(forceX, forceY));
```

```
body.velocity = new Vector2(body.velocity.x, forceY);
```





Bài tập

- Thiết kế nhân vật di chuyển sang trái – phải bằng keyboard
 - Thiết kế nhân vật nhảy, di chuyển nhảy lên các Platform
 - Thiết kế các nhân vật di chuyển qua lại trên màn hình, di chuyển qua lại trong một khoảng
- 