



Xử lý va chạm



# Một số hàm thường dùng

- ▶ `void OnTriggerEnter2D(Collider2D collision)`
- ▶ `void OnCollisionEnter2D(Collision2D collision)`




# Điều kiện bắt sự kiện OnCollisionEnter2D ở Object A

- Object A:
  - Collider Component
  - Is Trigger = False
  - Ít nhất 1 object có Rigidbody component
  - IsKinematic không được = True ở cả hai object
- Object B:
  - Có Collider Component
  - Is Trigger = False




## Điều kiện bắt sự kiện OnTriggerEnter2D ở Object A

- Cả 2 object có Collider Component
- Ít nhất 1 object có IsTrigger = True
- Ít nhất 1 object có Rigidbody component




# Va chạm vào vật biến mất

```
private void OnTriggerEnter2D(Collider2D collision)
{
    if(collision.tag=="player")
    {
        Destroy(collision.gameObject);
    }
}
```



# Demo: Xử lý bắn đạn

- Trúng đạn vật biển mất
- Đạn bay ra khỏi màn hình biển mất
- Đạn rớt xuống nền biển mất

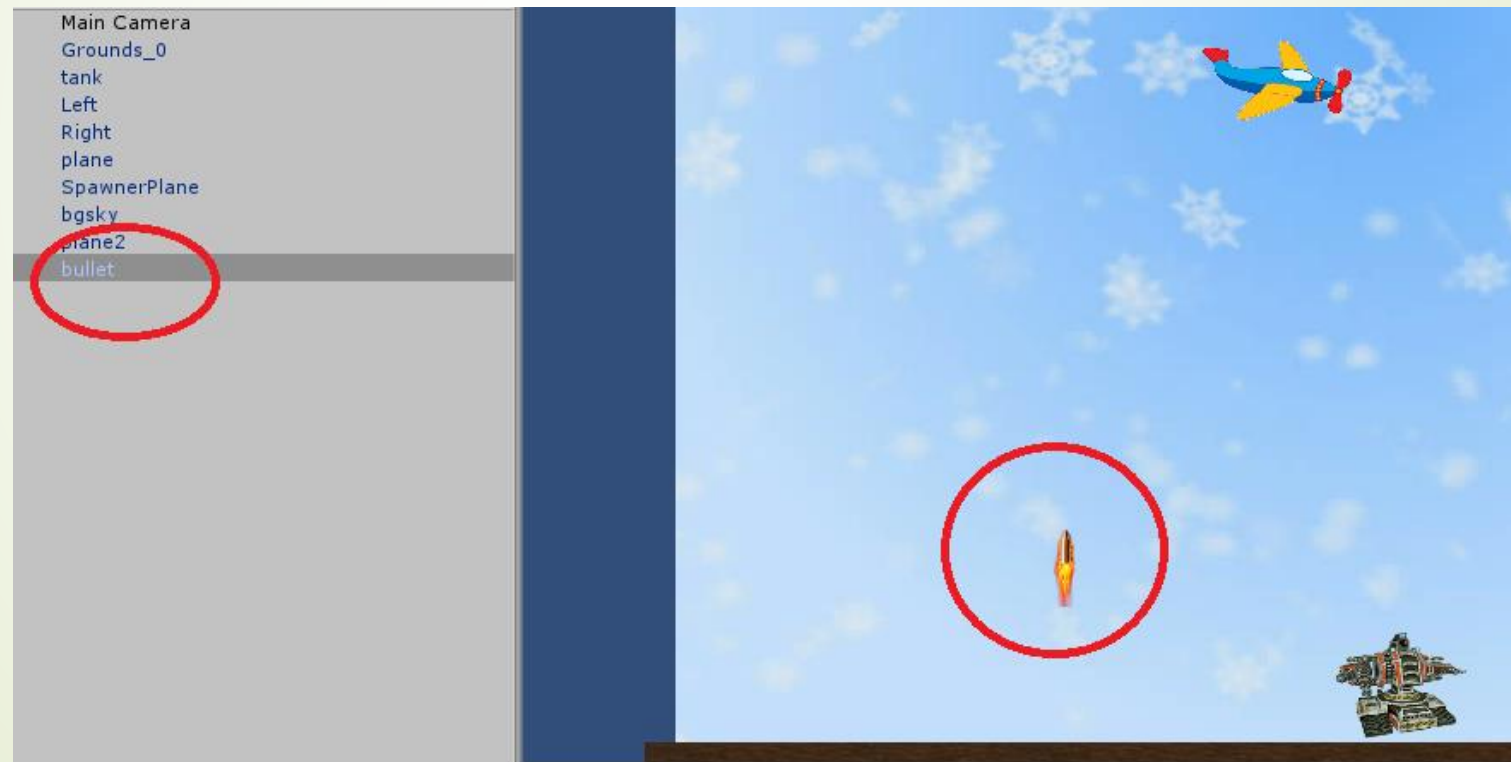


# Demo: Xử lý bắn đạn

```
if (collision.gameObject.tag == "plane"){  
    Destroy(gameObject);  
}  
//-----  
Vector2 temp = transform.position;  
temp.y += 10 * Time.deltaTime;  
transform.position = temp;  
if (temp.y > 6) Destroy(this.gameObject);
```

# Demo: Tạo gameobject Bullet

- Bước 1: Tạo gameobject bullet từ sprite





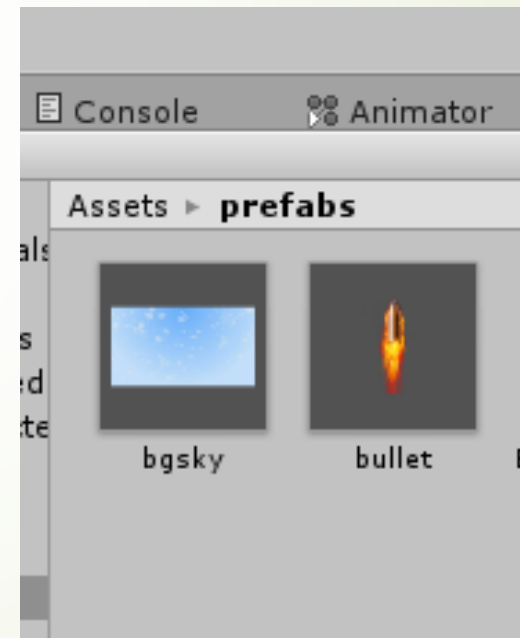
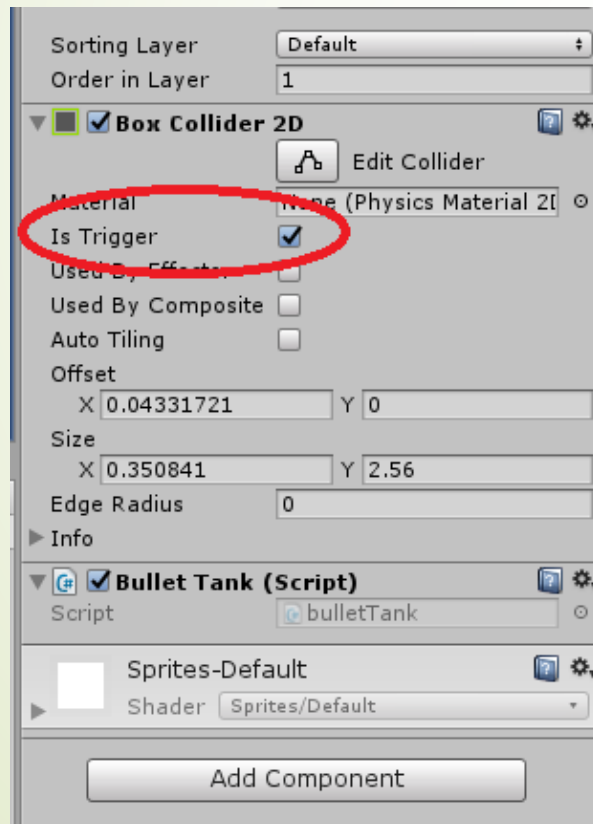
# Demo: Tạo gameobject Bullet

- Bước 2: Viết script cho Bullet

```
public class bulletTank : MonoBehaviour {  
  
    // phải sử dụng trigger cho bullet, nếu không sẽ đẩy bullet bật ngược  
  
    void Update () {  
        Attack();  
    }  
  
    void Attack()  
    {  
        Vector2 temp = transform.position;  
        temp.y += 20 * Time.deltaTime;  
        transform.position = temp;  
  
        if (temp.y > 6) Destroy(this.gameObject);  
    }  
  
    public void OnTriggerEnter2D(Collider2D collision)  
    {  
        if (collision.gameObject.tag == "plane")  
        {  
            Destroy(gameObject);  
        }  
    }  
}
```

# Demo: Tạo gameobject Bullet

- Bước 3: Add component script cho bullet, Add BoxCollider và set isTrigger = true
- Bước 4: và kéo bullet vào prefabs



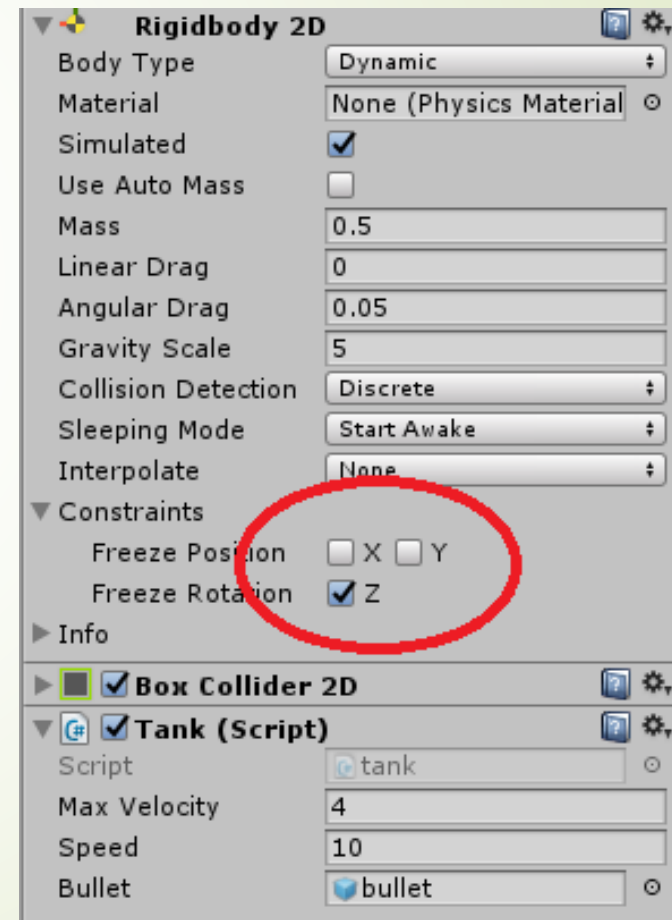
# Demo: điều khiển xe tăng

Bước 1: Thêm gameobject Tank, Plane



# Demo: điều khiển xe tăng


Bước 2: Thêm Rigidbody2D, Boxcollider2D





# Demo: điều khiển xe tăng

Bước 3: Viết script điều khiển Tank di chuyển sang trái phải và bắn



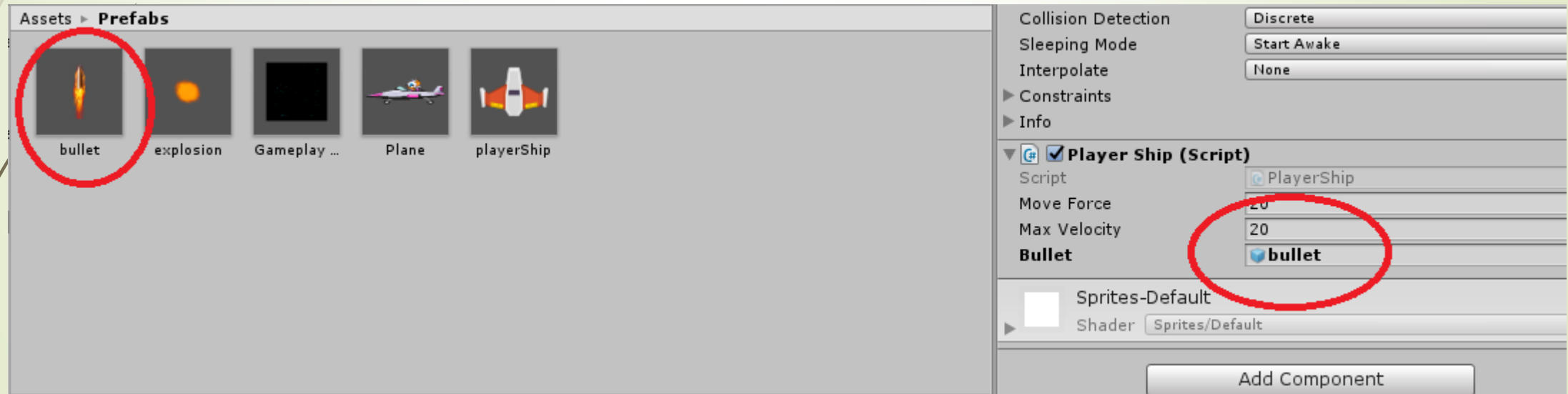
# Demo: điều khiển xe tăng

```
public class tank : MonoBehaviour {  
    public float maxVelocity = 4f;  
    public float speed = 10f;  
    private Rigidbody2D mybody;  
  
    [SerializeField]  
    private GameObject bullet;  
  
    void Awake () {  
        mybody = GetComponent<Rigidbody2D>();  
    }  
  
    void FixedUpdate()  
    {  
        Move();  
  
        if (Input.GetKeyUp(KeyCode.Space)) //bắn  
        {  
            Instantiate(bullet, transform.position, Quaternion.identity);  
        }  
    }  
}
```


```
private void Move()  
{  
    float forceX = 0f;  
    float vel = Mathf.Abs(mybody.velocity.x);  
  
    float h = Input.GetAxisRaw("Horizontal");  
  
    if (h > 0)  
    {  
        if (vel < maxVelocity) forceX = speed;  
    }  
    else if (h < 0)  
    {  
        if (vel < maxVelocity) forceX = -speed;  
    }  
  
    mybody.AddForce(new Vector2(forceX, 0));  
}  
}
```

# Demo: điều khiển xe tăng

- Kéo GameObject bullet từ Prefabs vào biến bullet







# Demo Plane 1: máy bay di chuyển qua lại trên màn hình, nổ nếu bị trúng đạn.


- ▶ Tạo script cho plane
  - ▶ Sự kiện OnTriggerEnter2D()



# Demo Plane 1: máy bay di chuyển qua lại trên màn hình, nổ nếu bị trúng đạn.


```
public class plane : MonoBehaviour {  
    private float speed=2f;  
    [SerializeField]  
    private GameObject effect;  
  
    // Update is called once per frame  
    void Update () {  
        Vector3 temp = transform.position;  
        temp.x -= speed * Time.deltaTime;  
        if (temp.x < -8)  
        {  
            temp.x = -8;  
            speed = -speed;  
        }  
  
        if(temp.x>8)  
        {  
            temp.x = 8;  
            speed = -speed;  
        }  
  
        transform.position = temp;  
    }  
}
```

```
private void OnTriggerEnter2D(Collider2D collision)  
{  
    if (collision.gameObject.tag == "bullet")  
    {  
        Destroy(gameObject);  
        Destroy(Instantiate(effect, transform.position, this.  
transform.rotation),2);  
    }  
}
```



## Demo Plane 2: máy bay di chuyển chạm đường biên quay lại dùng GameObject Left, Right

- ▶ Tạo 2 GameObject Left và Right đặt ở bên trái và phải màn hình có BoxCollider, check IsTrigger2D
- ▶ Tạo script cho plane
  - ▶ hành vi Move()
  - ▶ Sự kiện OnTriggerEnter2D()



## Demo Plane 2: máy bay di chuyển chạm đường biên quay lại dùng GameObject Left, Right


```
void Move()
{
    Vector2 temp = transform.position;
    temp.x -= speed * Time.deltaTime;

    transform.position = temp;
}
```

## Demo Plane 2: máy bay di chuyển chạm đường biên quay lại dùng GameObject Left, Right

```
private void OnTriggerEnter2D(Collider2D collision)
{
    if(collision.gameObject.tag == "left")
    {
        speed = -speed;
        Vector2 scale = transform.localScale;
        scale.x = -1f;
        transform.localScale = scale;
    }
}
```

```
if(collision.gameObject.tag == "right")
{
    speed = -speed;
    Vector2 scale = transform.localScale;
    scale.x = 1f;
    transform.localScale = scale;
}
```



## Demo Plane 3: máy bay di chuyển từ phải sang trái, chạm đường biên biến mất

- ▶ Tạo 2 GameObject Left và Right đặt ở bên trái và phải màn hình có BoxCollider, check IsTrigger2D
- ▶ Tạo script cho plane
  - ▶ hành vi Move()
  - ▶ Sự kiện OnTriggerEnter2D()
- ▶ Tại một GameObject SpawnerPlane cho phép tự sinh Plane sau 2-3 giây và random vị trí xuất hiện

# Demo Plane 3: máy bay di chuyển từ phải sang trái, chạm đường biên biến mất

➡ Viết script cho GameObject SpawnerPlane

```
public class SpawnerPlane : MonoBehaviour {

    [SerializeField]
    private GameObject plane;

    // Use this for initialization
    void Start () {
        StartCoroutine(CreatePlane());
    }
    IEnumerator CreatePlane()
    {
        yield return new WaitForSeconds(2);

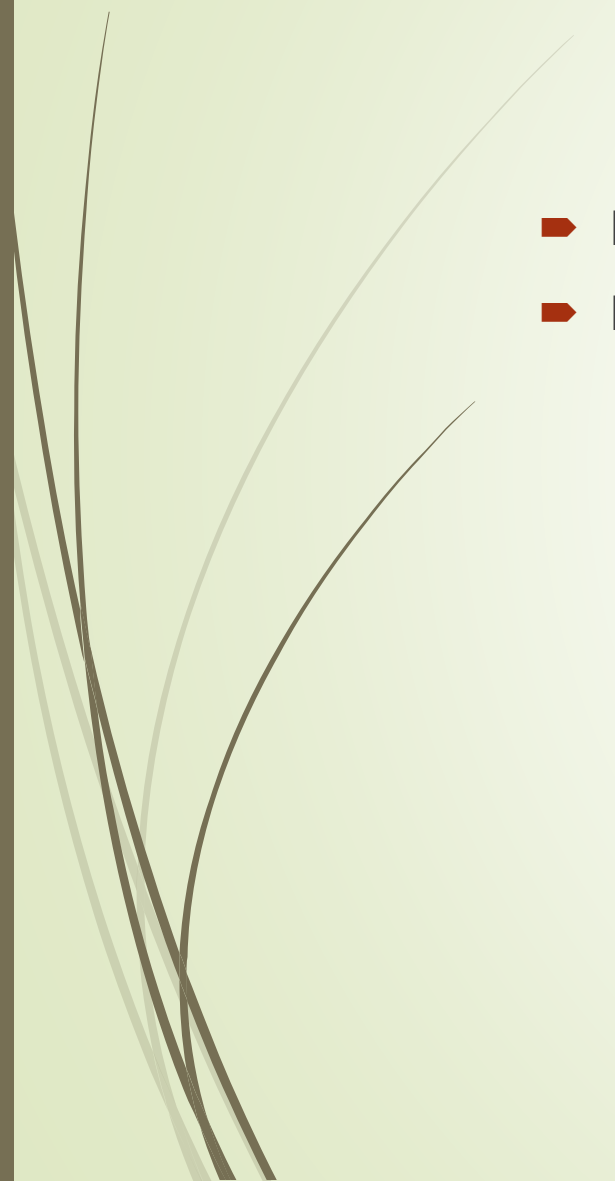
        Vector2 temp = transform.position;
        temp.y += Random.Range(-2, 2);

        Instantiate(plane, temp, this.transform.rotation);

        StartCoroutine(CreatePlane());
    }
}
```



# Demo: Xử lý vụ nổ

- Dùng hiệu ứng particle
  - Dùng animation
- 



# Hiệu ứng Particle

- ▶ Menu GameObject – Particle System
- ▶ Thêm 1 component GameObject vào script cần có hiệu ứng
- ▶ Tạo đối tượng hiệu ứng particle

*Destroy(Instantiate(exclusionEffect, transform.position, this.transform.rotation), 2);*



# Hiệu ứng Animation

- ▶ Thêm animation cho vụ nổ
- ▶ Thêm 1 component GameObject vào script cần có hiệu ứng
- ▶ Tạo đối tượng animation khi nổ

*Destroy(Instantiate(boom, transform.position, this.transform.rotation), 2);*