Introduction to Unity 2017/18

Demystifying Game Prototyping
by Peter Bickhofe, November 2017

So, you want to be game developer?

Congrats!
You're in good company.

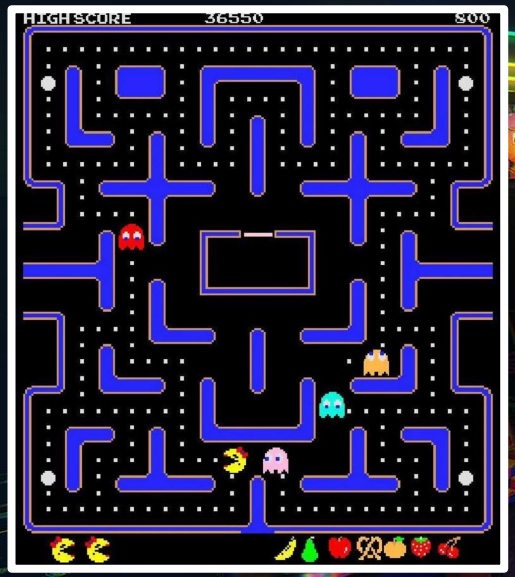

NOOB PROGRAMMERS
NOOB PROGRAMMERS EVERYWHERE
memegenerator.net

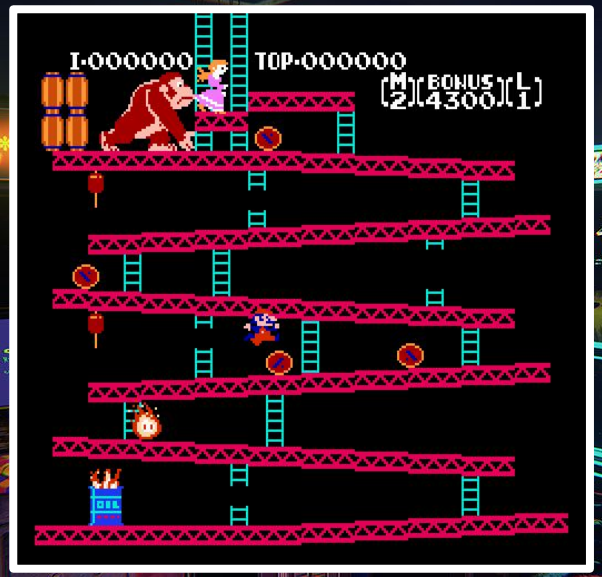
Let met tell you a secret...

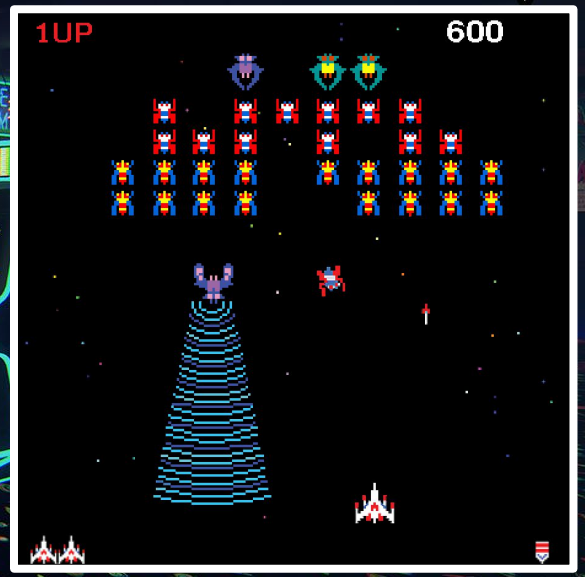
All classic* games have three basic things in common...

*some contemporary games, too!

Three different games from the 80s

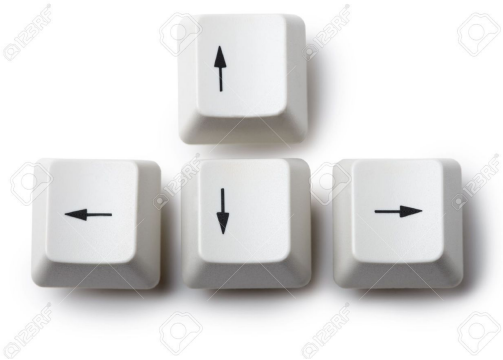






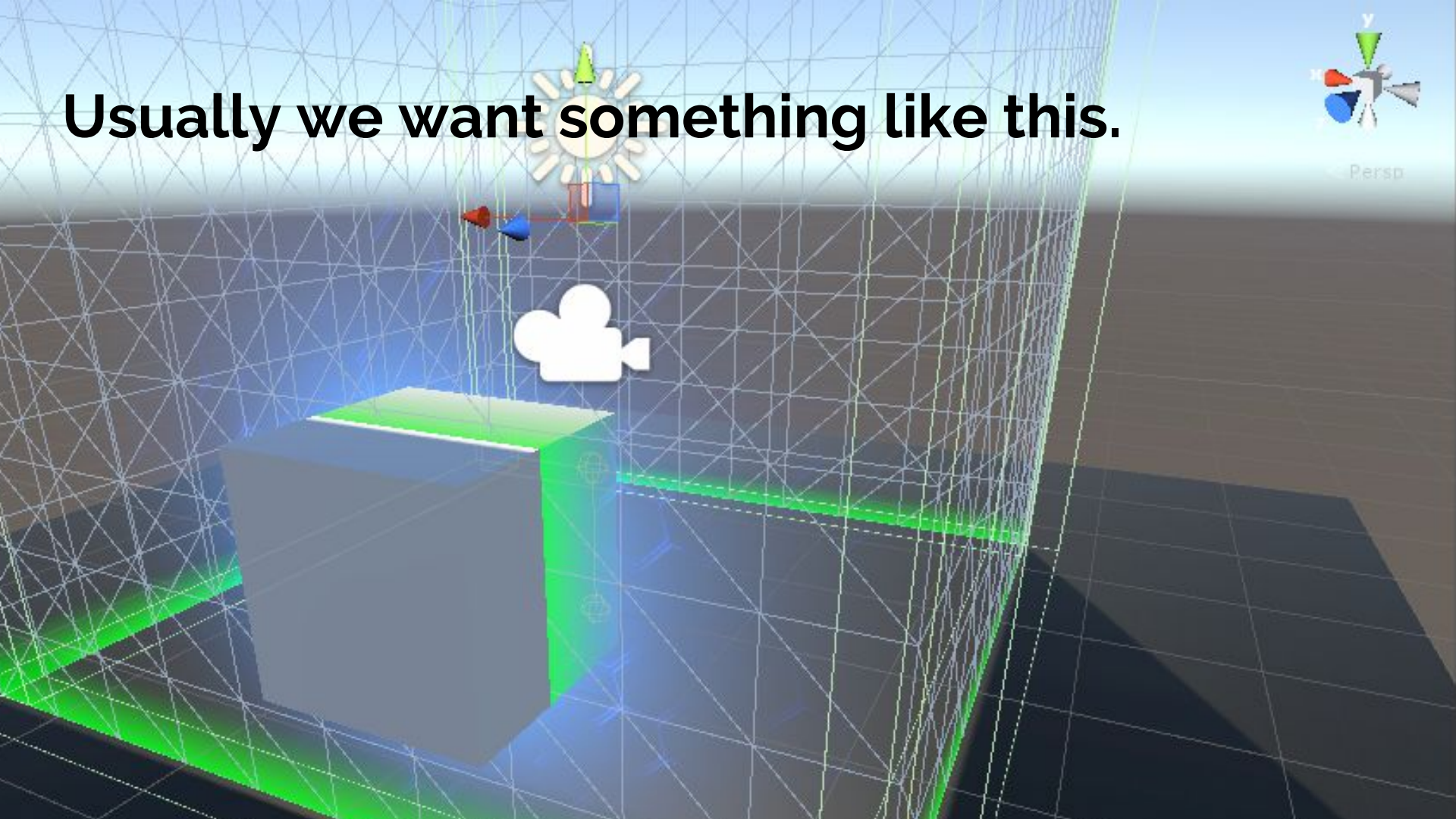
1. Input/Control

Input: Keyboard, Mouse & Touch



2. Collision

Not necessarily this kind of...


Usually we want something like this.
y
Persp

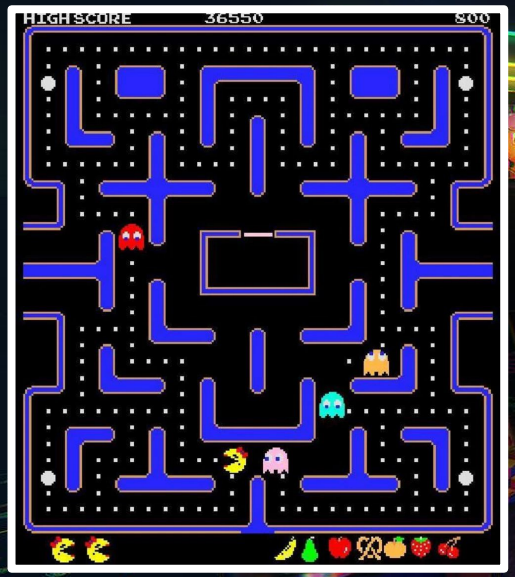
3. Instantiation

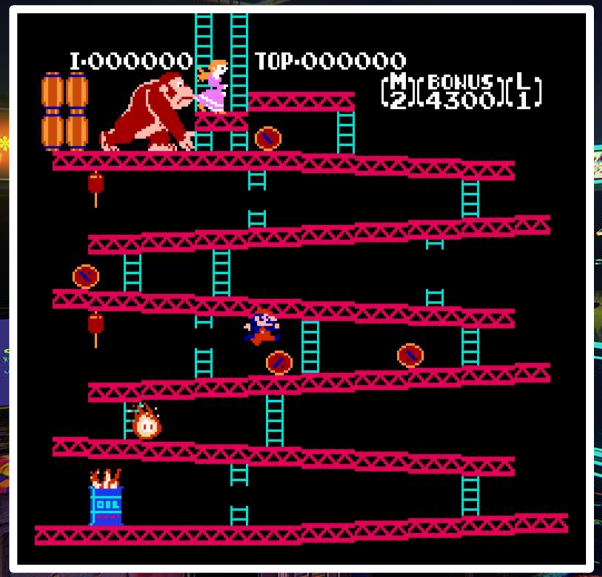
Tataaaaa...

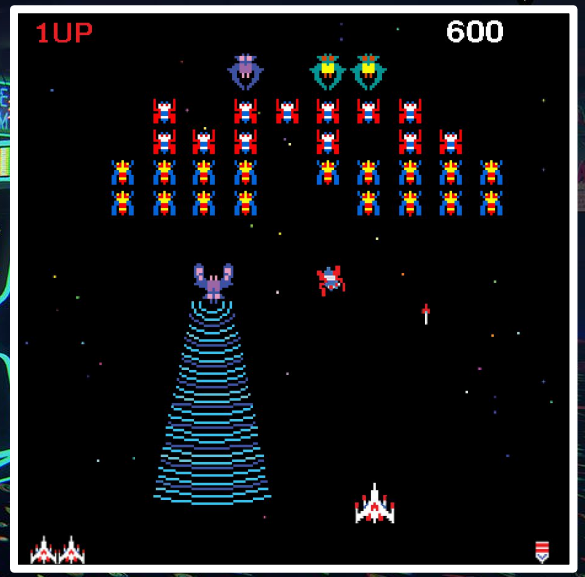
End of lecture

#not

Maybe some similar mechanics?







Input: Keyboard

```
using UnityEngine;
using System.Collections;

public class ExampleClass : MonoBehaviour
{
    public void Update()
    {
      if (Input.GetKey(KeyCode.UpArrow))
      {
            Debug.Log("Up arrow pressed!");
      }
    }
}
```

Input: Mouse Button

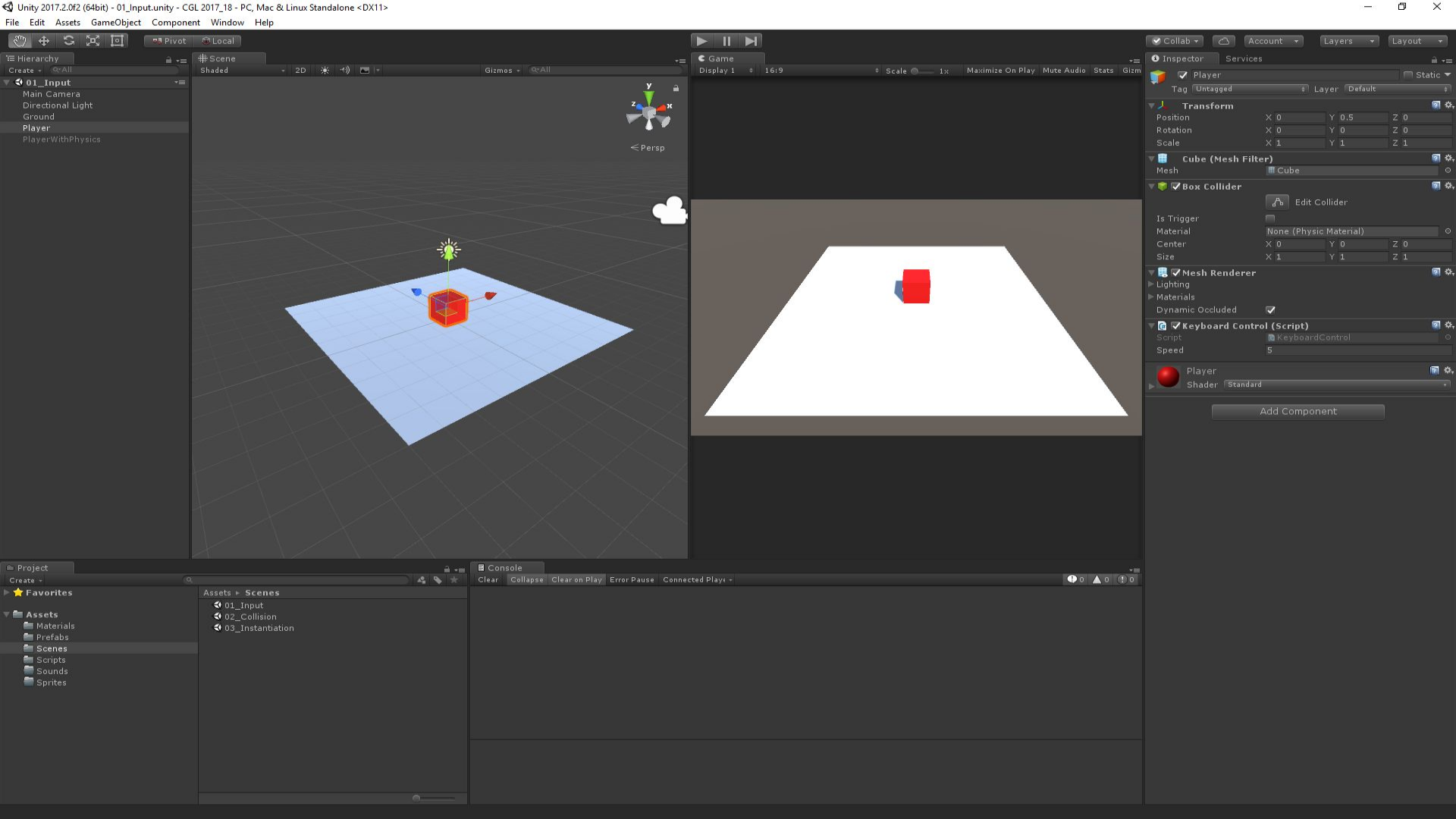
```
using UnityEngine;
using System.Collections;

public class ExampleClass : MonoBehaviour
{
    public void Update()
    {
      if (Input.GetMouseButton(0))
      {
            Debug.Log("Pressed left click.");
      }
    }
}
```

Input: Touch

```
using UnityEngine;
using System.Collections;

public class ExampleClass : MonoBehaviour
{
    public void Update()
    {
      if (Input.touchCount > 0 && Input.GetTouch(0).phase == TouchPhase.Began)
      {
            Debug.Log("Touched!");
      }
    }
}
```


Unity 2017.2.0f2 (64bit) - 01_Input.unity - CGL 2017_18 - PC, Mac & Linux Standalone <DX11>
File Edit Assets GameObject Component Window Help
Pivot Local
Hierarchy
Create
01_Input
Main Camera
Directional Light
Ground
Player
PlayerWithPhysics
Scene
Shaded 2D Gizmos
Persp
Game
Display 1 16:9 Scale 1x Maximize On Play Mute Audio Stats
Collab Account Layers Layout
Inspector Services
Player
Static
Tag Untagged Layer Default
Transform
Position X 0 Y 0.5 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
Cube (Mesh Filter)
Mesh Cube
Box Collider
Edit Collider
Is Trigger
Material None (Physic Material)
Center X 0 Y 0 Z 0
Size X 1 Y 1 Z 1
Mesh Renderer
Lighting
Materials
Dynamic Occluded
Keyboard Control (Script)
Script KeyboardControl
Speed 5
Player
Shader Standard
Add Component
Project
Create
Favorites
Assets
Materials
Prefabs
Scenes
Scripts
Sounds
Sprites
Assets ▸ Scenes
01_Input
02_Collision
03_Instantiation
Console
Clear Collapse Clear on Play Error Pause Connected Player

Input: Movement

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

public class KeyboardControl : MonoBehaviour {

	public float speed = 5.0f;

	void Update ()
	{
		if (Input.GetKey(KeyCode.UpArrow))
		{
			Debug.Log("Up arrow pressed!");
			transform.Translate(Vector3.forward * speed * Time.deltaTime);
		}
	}
}
```

Input: Movement with Physics

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

public class KeyboardControlWithPhysics : MonoBehaviour {

	public float force = 5.0f;
	public Rigidbody rb;

	void Update ()
	{
		if (Input.GetKey(KeyCode.UpArrow))
		{
			print("up");
			rb.AddForce(Vector3.forward * force);
		}
	}
}
```

Collision (by Hand)

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

public class CheckBorder : MonoBehaviour {

	void Update () {
		if (transform.position.x > 4) print (“stop”);
	}
}
```

Collisions (Physics)

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

public class CheckCollision : MonoBehaviour {

	Rigidbody rb;

	void OnCollisionEnter(Collision collision)
	{
		print("hit: " + collision.gameObject.name);
	}

}
```

Colliders and Triggers (Physics)

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

public class CheckCollision : MonoBehaviour {

	Rigidbody rb;

	void OnTriggerEnter(Collider collider)
	{
		print("Enter: " + collider.gameObject.name);
	}

}
```

Spawn objects

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

public class SpawnObject : MonoBehaviour {

	public GameObject Pill;

	void Start()
	{
		if (Input.GetKeyDown(KeyCode.Space))
		{
			Instantiate(Pill, transform.position, Quaternion.identity);
		}
	}
}
```

Fire bullet (cannonball)

New object has physics/rigidbody!

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

public class FireCannonball : MonoBehaviour {

	public GameObject Bullet;
	public float Power = 1000f;

	void Update ()
	{
		if (Input.GetMouseButtonDown(0))
		{
		GameObject NewBullet = Instantiate(Bullet, Vector3.zero,
		Quaternion.identity);

		NewBullet.GetComponent<Rigidbody>().AddForce(Vector3.forward * Power);
	}
}
```

Three tools to make a game!

Your task

Create a game based on the principles of
“Input”, “Collision” and “Instantiation”.

Start with a scribble that fits on one DIN A4 sheet.

Github
https://github.com/bickhofe/CGL-2017-18

Part two

Some additional basics

Working with scenes

Canvas, Textfields and 3DText

Adding UI Buttons

Adding Bitmaps/Sprites

PlayerPrefs

Sound

Your task for 15th January

Finish “your” game with:
- **Scenes** (Title, Game, Results)
- **Scores or text output**
- **Sound**

Github
https://github.com/bickhofe/CGL-2017-18