GameDev.tv

Unity 2021.1

Snow Boarder Section

Rick Davidson

GameDev.tv

Unity 2021.1

Section Intro - Snow Boarder

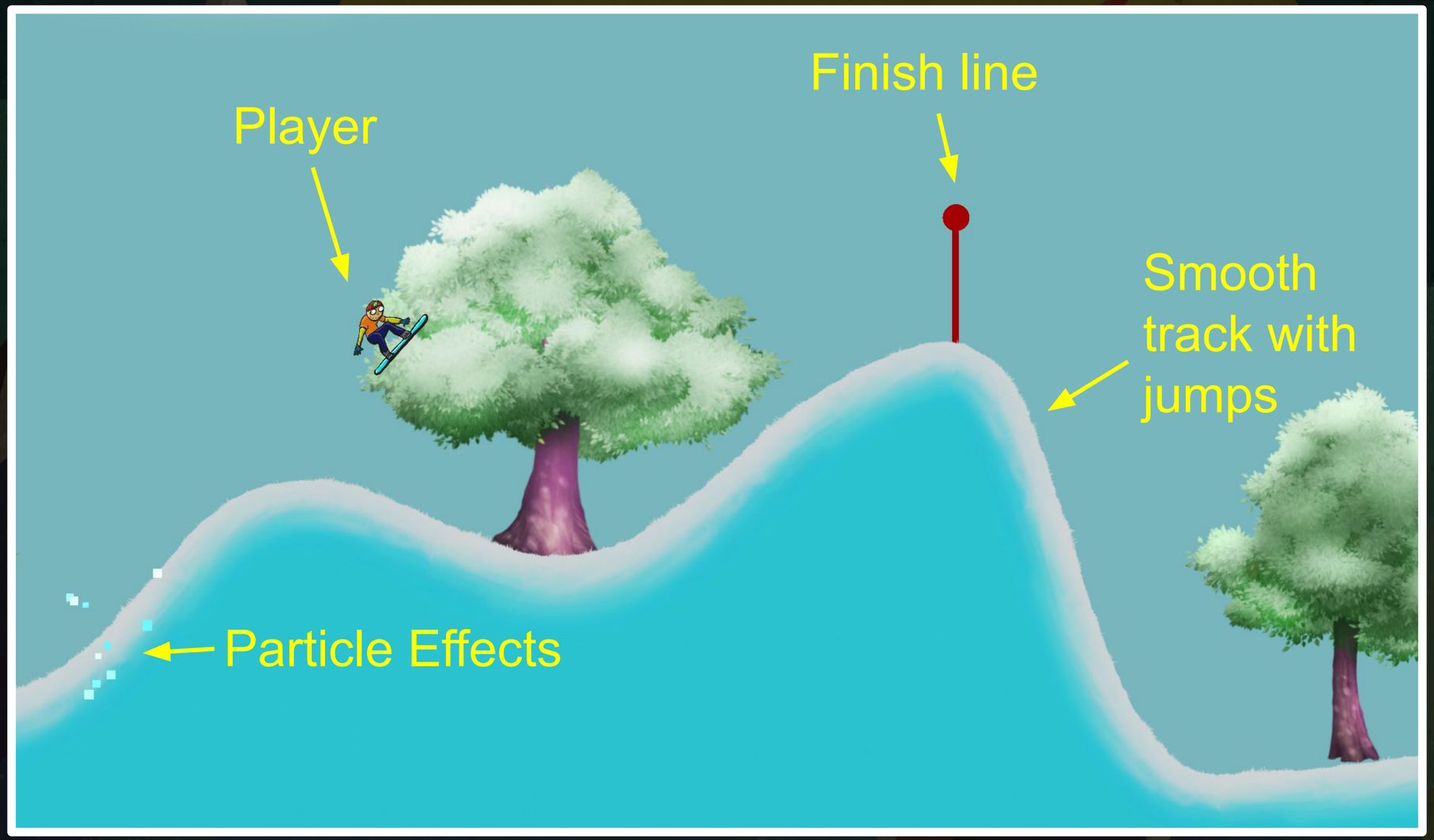
GameDev.tv

Unity 2021.1

Game Design - Snow Boarder

Rick Davidson

Gameplay Overview Screen



Game Mechanics We Need

- Move along the track
- Rotate forwards and backwards
- Ability to speed up
- Particle effects that only play when we’re touching ground
- Finish line that restarts level
- Crash detection which restarts the level

Game Design

Player Experience:

Smooth, relaxing

Core Mechanic:

Don’t crash

Game Loop:

Reach the end to win

A Quick Challenge

- Name your player character
- I’ll be naming mine Barry
- Let us know your choice in the discussions

GameDev.tv

Unity 2021.1

How To Use Sprite Shapes

Rick Davidson

Make A Simple Level

- Modify your sprite shape to create a simple level layout
- Don’t spend too much time yet, just something to let us test our gameplay

GameDev.tv

Unity 2021.1

Using Edge Colliders

Rick Davidson

Quick Challenge

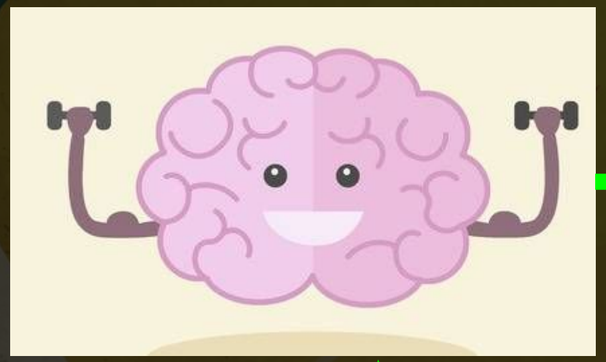
- Make a ball roll down the hill


GameDev.tv
Unity 2021.1
Add Cinemachine Follow Camera

What Is Cinemachine?

- **Cinemachine** is a powerful package that lets us:
  - manage multiple cameras in our scene
  - Easily create rules for our cameras

Cinemachine Brain



Main Camera

Virtual Cameras

Add Cinemachine

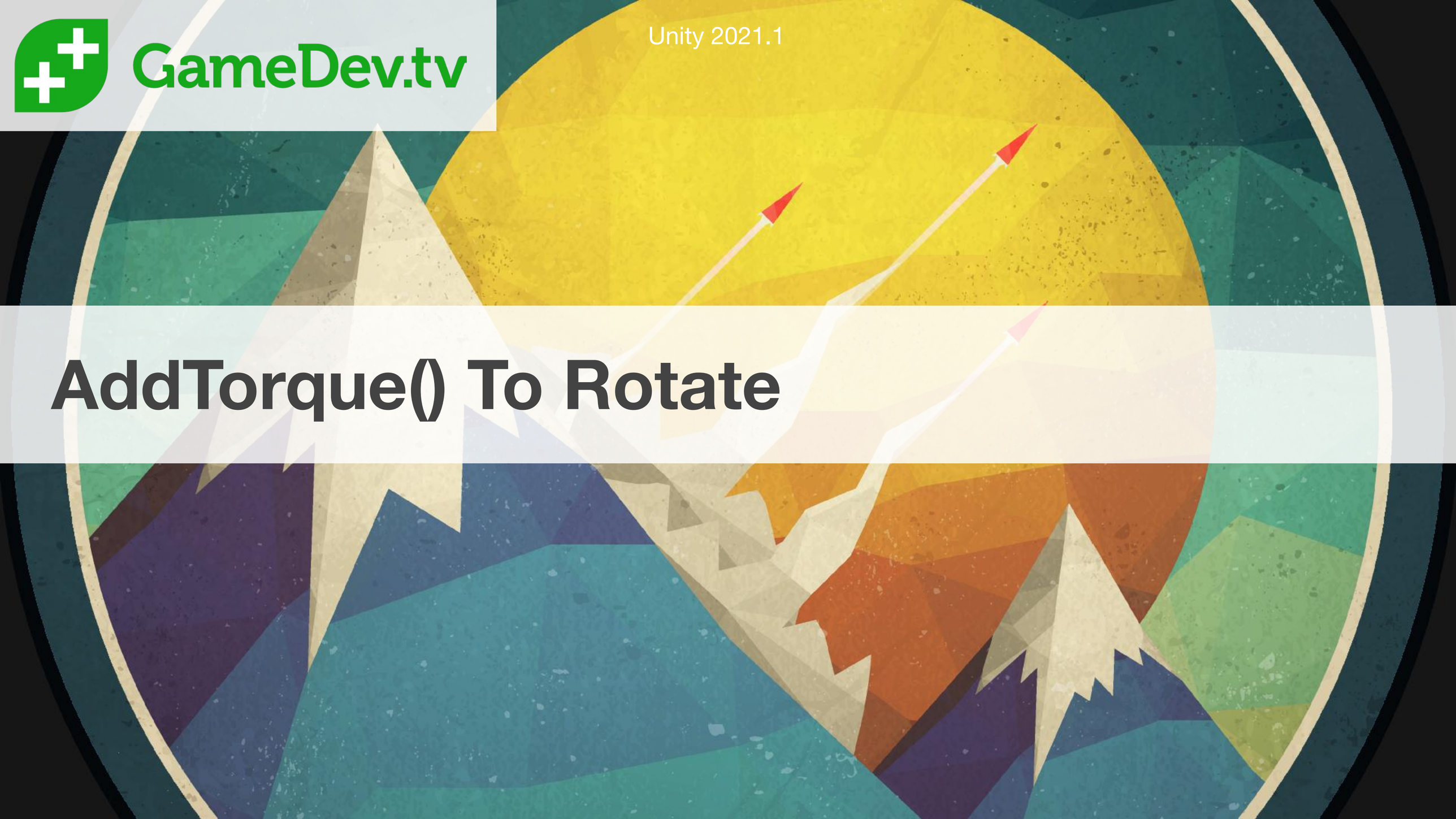
- Add the Package Manager window
- Find and install Cinemachine
- Add a Virtual Camera
- Point it to follow the ball
- Change the Screen X value to show more of whats to come
- Feel free to play around with the other settings

Set Up Our Character

Add A Character

- Either use our character or download / make your own
- Add the character in place of the ball
- Add the components you think the character needs so that its ready to be a snowboarder


GameDev.tv
Unity 2021.1
Using Surface Effector 2D


GameDev.tv
Unity 2021.1
AddTorque() To Rotate

Rotate Our Character

- When the player pushes right arrow, make the character rotate the other direction


GameDev.tv
Unity 2021.1
Triggers To Restart Level

Finish The Crash Detector

- In our CrashDetector.cs script, write some code that will print to screen if we bonk our head on the ground.


GameDev.tv
Unity 2021.1
NameSpaces & SceneManagement

Organising Code

- The more code we have, the higher likelihood of conflicts between names and behaviours. Especially on multi-person projects.
  - Eg. Programmer 1 working on player uses “Movement” method but Programmer 2 working on enemy also uses “Movement” method.
- So in C# our code is grouped with a particular structure to reduce conflicts.

How C# Is Broadly Organised

```
namespace UnityEngine
    class OurClass
        SomeMethod()
            statementA;
            statementB;
```

What Are Namespaces?

- We use Namespaces to group together similar Classes of code (and Classes are containers for methods and variables).
- If we want to use a particular Class then we need to tell Unity which Namespace it belongs to with the `using` keyword.

SceneManagement Namespace

- We’re going to use classes and methods from the `SceneManagement` namespace
  - Pre-built functionality to help us load scenes

Finish The Crash Detector

- Reload our scene when we bump into the ground.


GameDev.tv
Unity 2021.1
Using Invoke() For Delays

Creating A Delay

- There are 2 useful approaches to “waiting a moment”.
  - Invoke
  - Coroutines
- Invoke is a bit easier to understand but not as powerful.
- It also uses string reference which are clumsy.

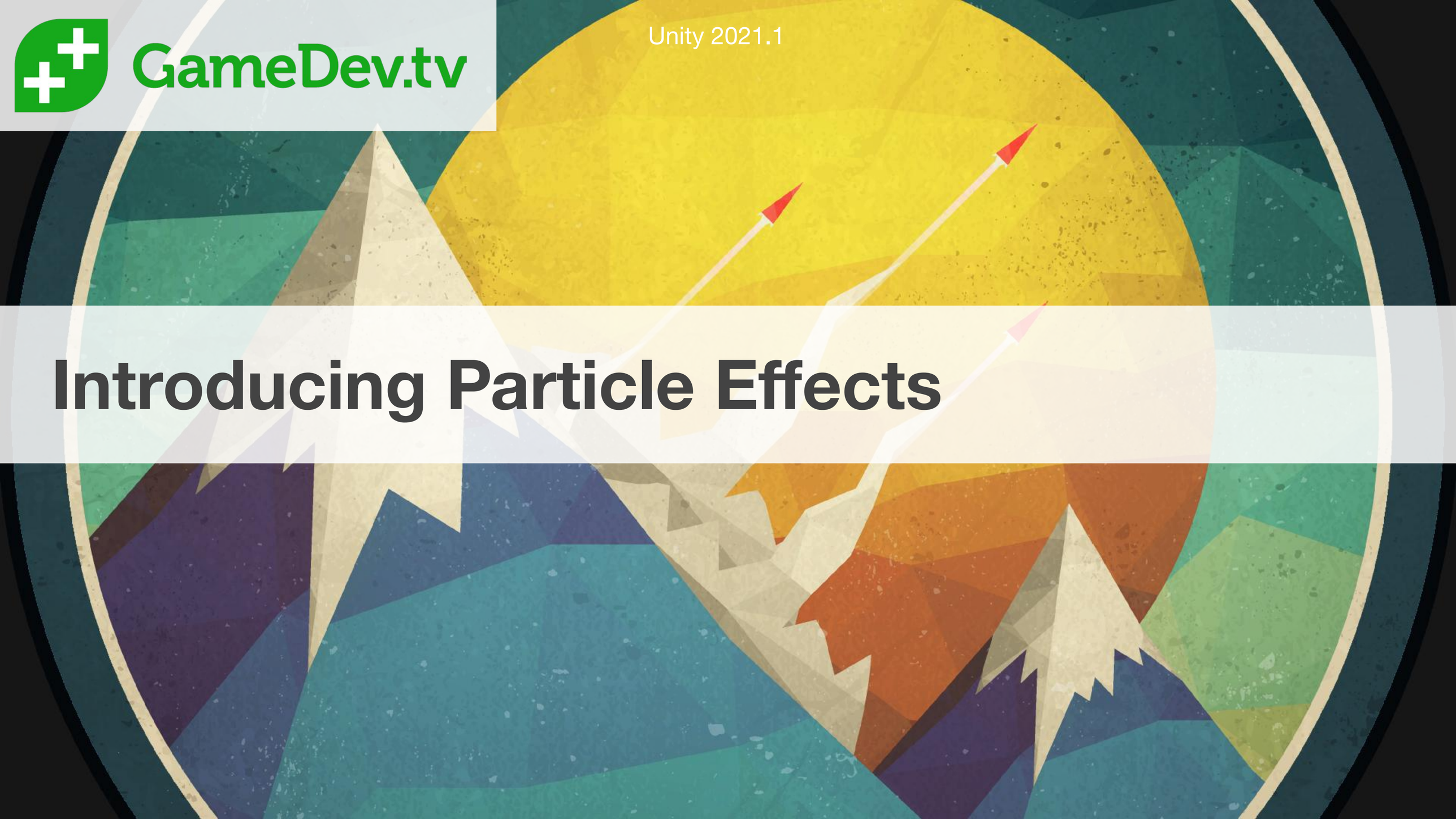
Using Invoke()

- When we call `Invoke()` we need to pass in the name of the method we want to call after a delay, as well as how long the delay is.

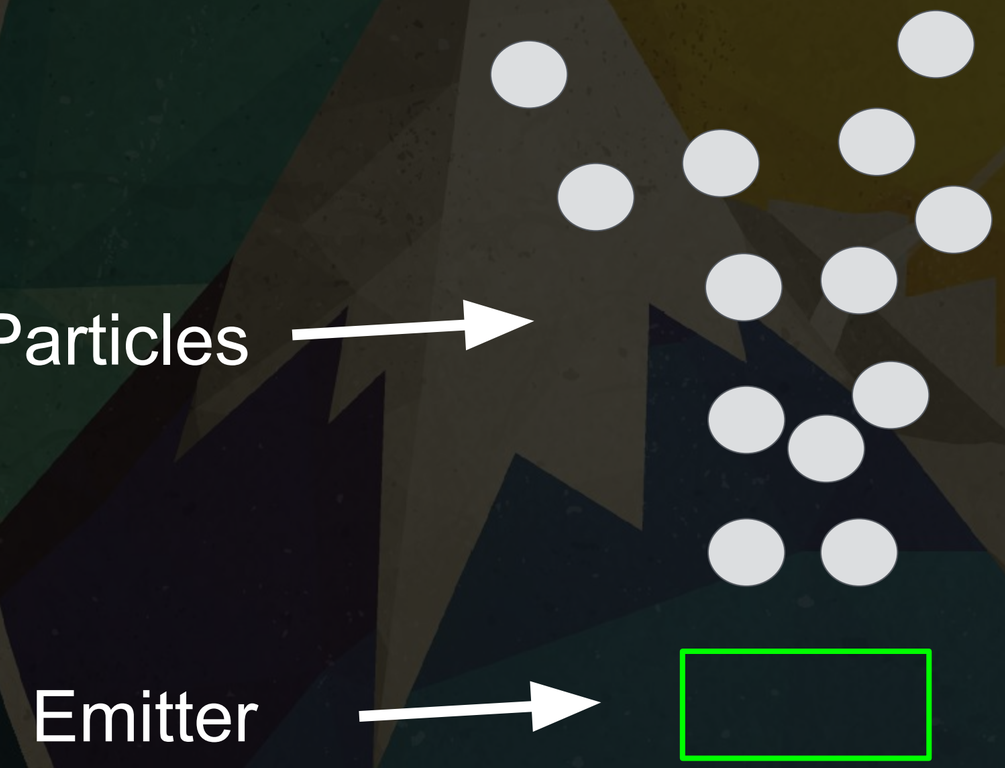
```
Invoke(“NameOfMethod”, delay);
```

Finish Our Invoke

- Instead of using a “magic number”, serialize a variable to represent our delay.
- Set up Invoke (with serialized variable) for our Crash Detection as well.


GameDev.tv
Unity 2021.1
Introducing Particle Effects

Particles System Component



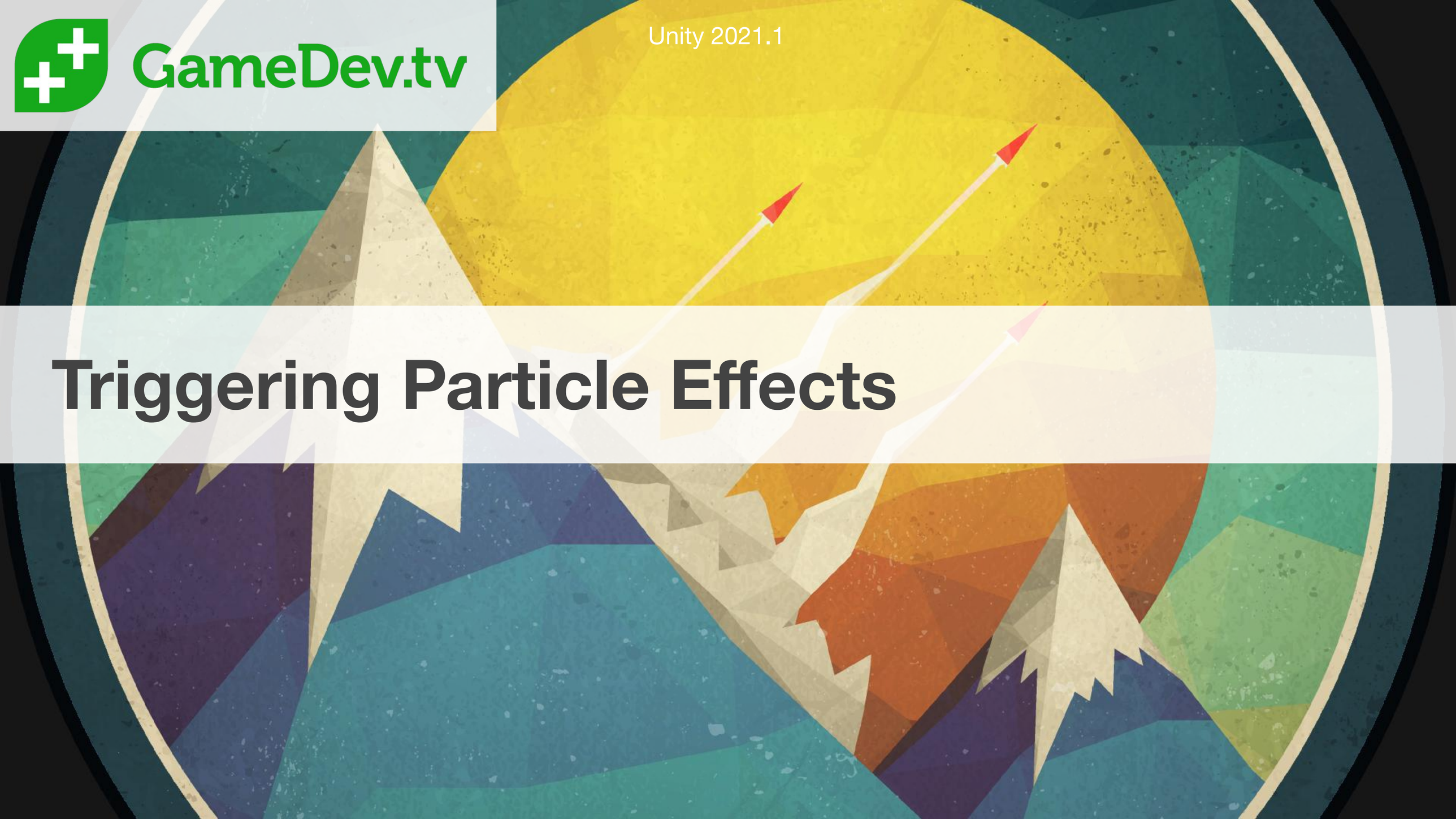
Particle System is a Component added to a Game Object

We use Modules for controlling behaviour

Each particle is not a Game Object

Create A Second Particle Effect

- Create a particle effect that we can trigger when the player crashes


GameDev.tv
Unity 2021.1
Triggering Particle Effects

Trigger Our Crash Particles

- When the player crashes, trigger the crash particles to play.

Take A Moment To Tune

Important To Tune Now And Then

- Keep your game “playable” as much as possible
- Improve the look and feel of your on regular basis
- This helps:
  - Keep you motivated
  - Show your game and get feedback
  - Generate ideas for improving your game

Iterate On What You Have

- Tune the player to improve the feel
  - Gravity, rotation speed, movement speed, etc
- Tweak the camera distance
- Add some background sprites (eg. trees, clouds)


GameDev.tv
Unity 2021.1
Using FindObjectOfType

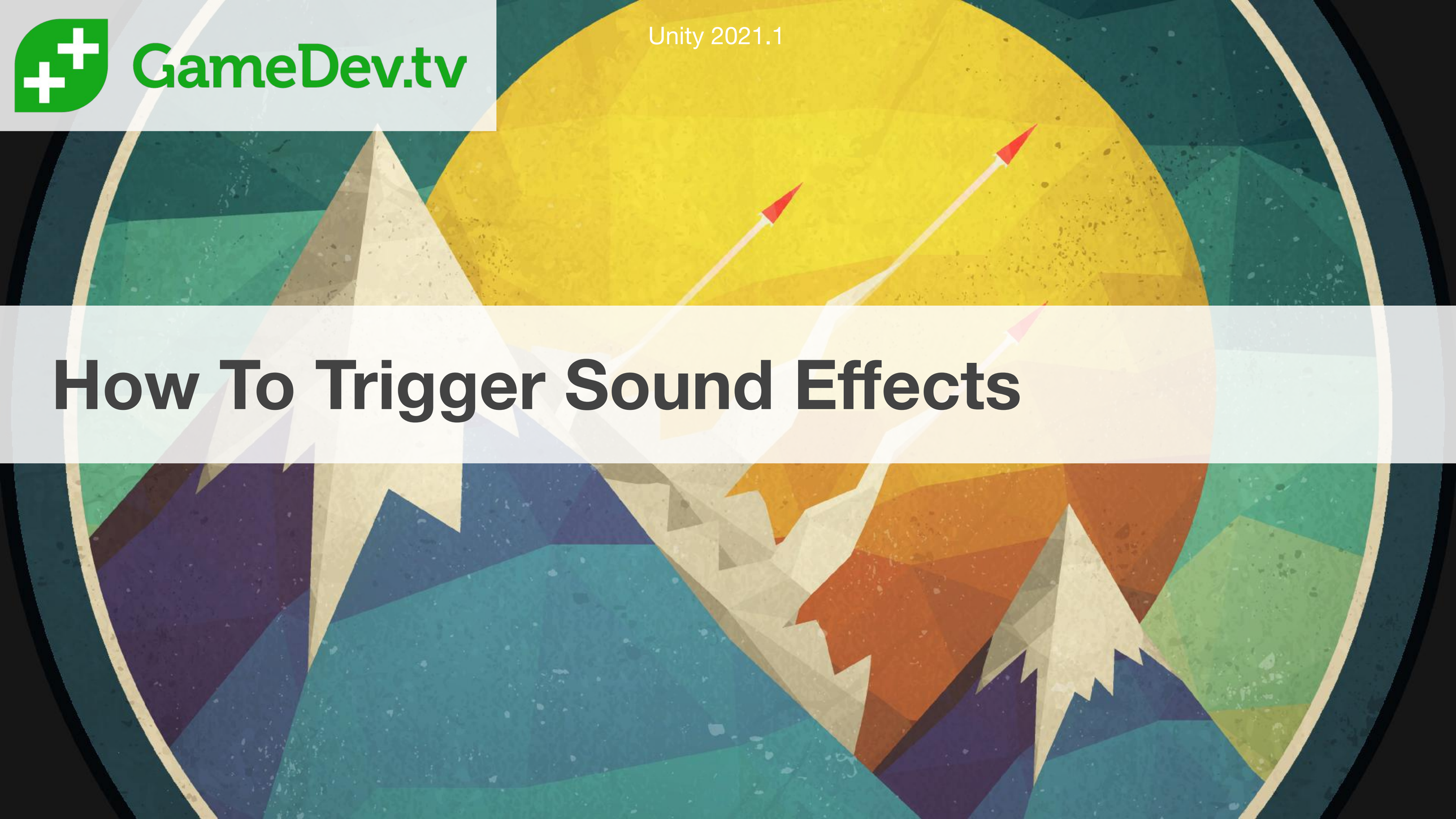
Make The Player Boost

- Finish our code so that you can boost (speed up) your player by increasing the speed of our effector.
- You'll need an if statement and else statement
- You'll need to get the player input
- You'll need to change the speed


GameDev.tv
Unity 2021.1
Using OnCollisionExit2D()

A Bigger Challenge

- Create a new script called DustTrail.cs
- Make particles that can be scattered along behind the player
- Only play the particles when the player is touching the ground
- Hints:
  - Consider using OnCollisionEnter2D and OnCollisionExit2D
  - For collision events we can use other.gameObject.tag
  - Consider using ParticleSystem.Stop()


GameDev.tv
Unity 2021.1
How To Trigger Sound Effects

Audio Terminology

- **Audio Listener** - like a microphone, receives sounds and plays through your computer's speakers
- **Audio Source** - Plays audio and allows us to adjust settings (eg. volume)
- **Audio Clip** - Contains the audio data to be played (mp3, Wav, OGG)

Grab A Couple Of SFX

- We need 2 sound effects (SFX)
  - Crash
  - Finish
- Find and download, or make 2 sound effects that we can use
- Or just download the ones I’ve provided for you


GameDev.tv
Unity 2021.1
Public Access Modifier

Prevent Double Play

Stop Double Plays

- Implement some code so that the crash SFX and particle effect can only be triggered once.
- Hints:
  - Consider using a bool
  - Consider using the && operator to see if something AND something is true

Create A Second Enemy

- Use the same structure from Enemy #1 to create a second Enemy
- Remember to use a prefab
- Tune the second Enemy so it has different hitpoints and strength
- Test that the difficulty level is good