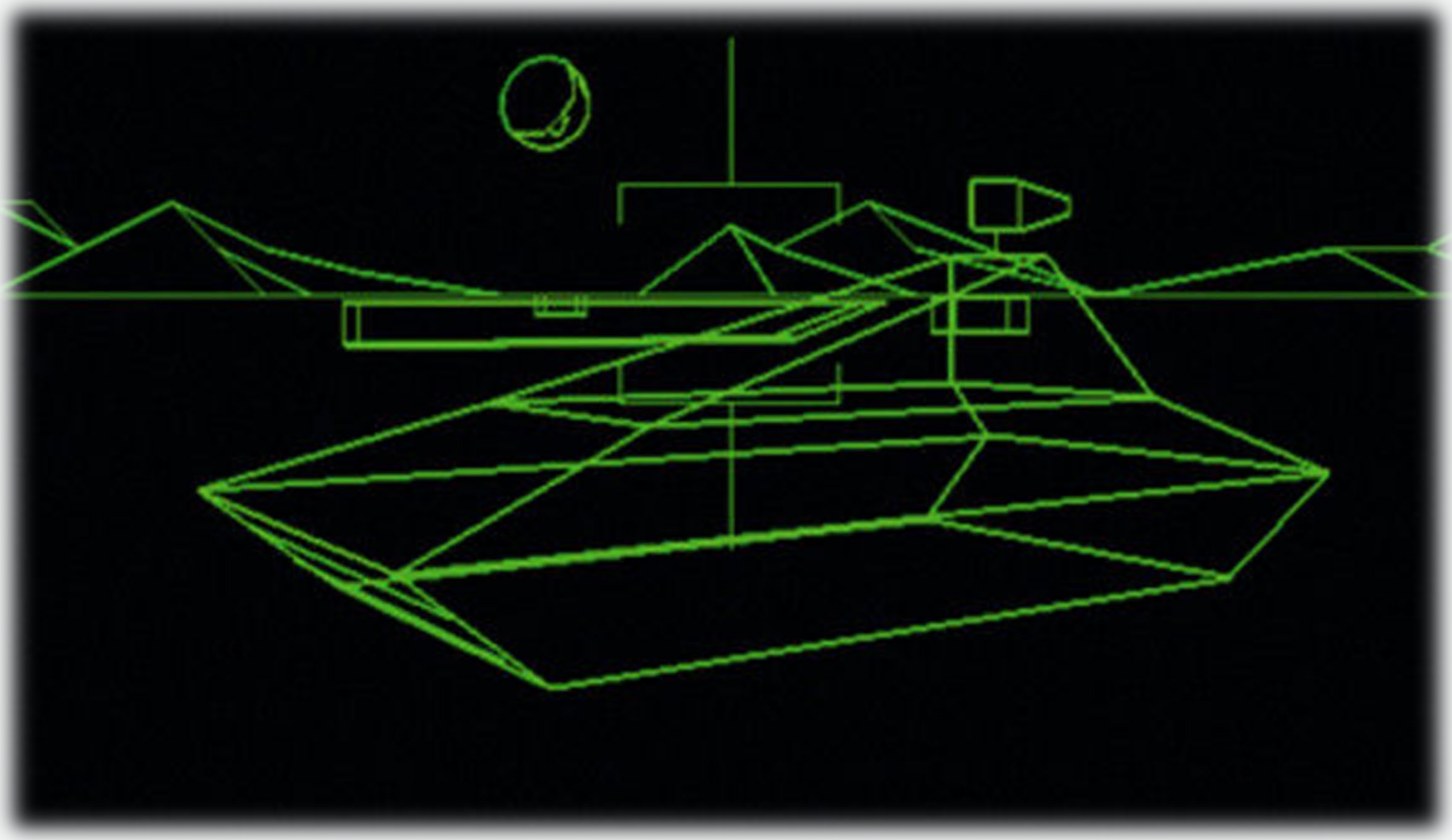
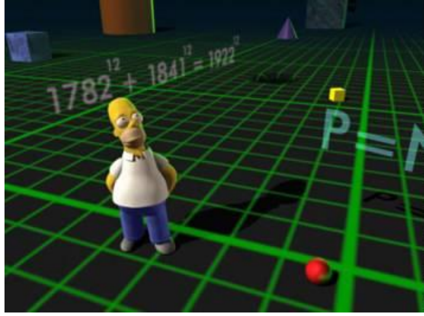


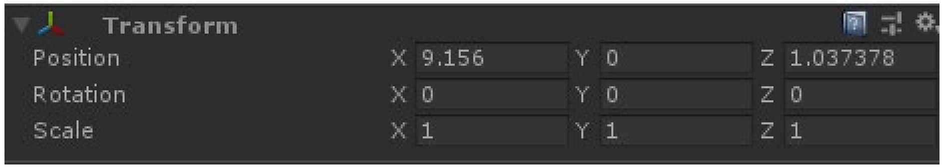
The 3rd Dimension

Entering the 3rd Dimensio

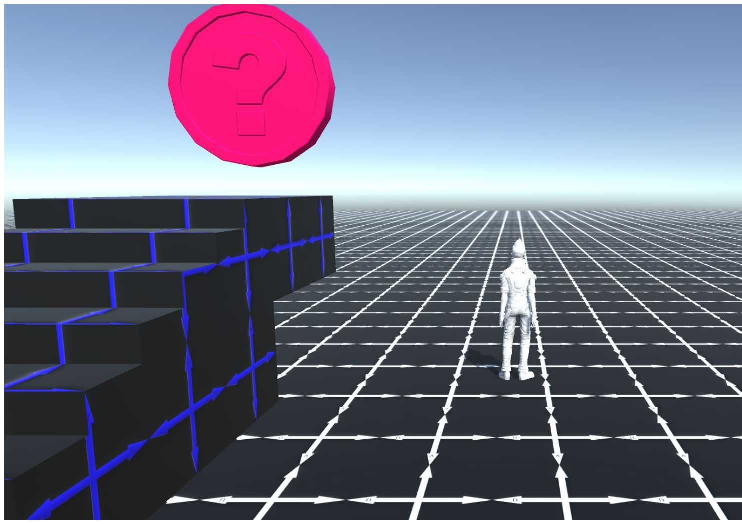


- Computer displays are 2D
- Perspective camera, uses a transform matrix to convert 3D vertices to 2D screen coordinates
- World 3D (x,y,z) ➔ Screen 2D (x/zdist , y/zdist)

- Every Object has a transform
  - Position
  - Rotation
  - Scale



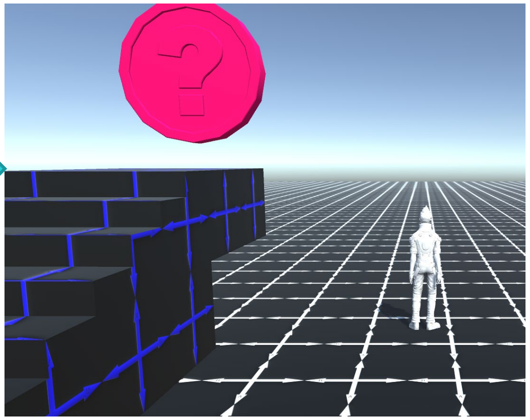
```
Transform
Position      X 9.156   Y 0   Z 1.037378
Rotation      X 0       Y 0   Z 0
Scale         X 1       Y 1   Z 1
```

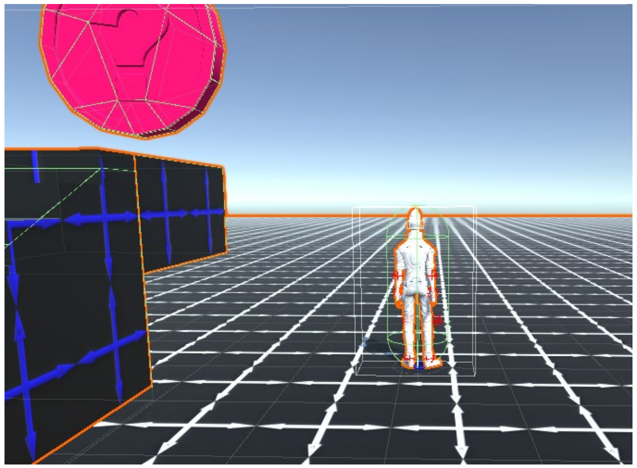


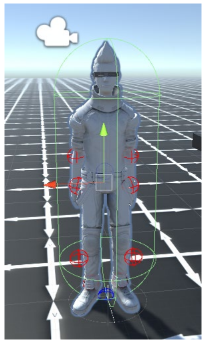
Game play VS GFX

Active game play elements and GFX are not the same

The player sees



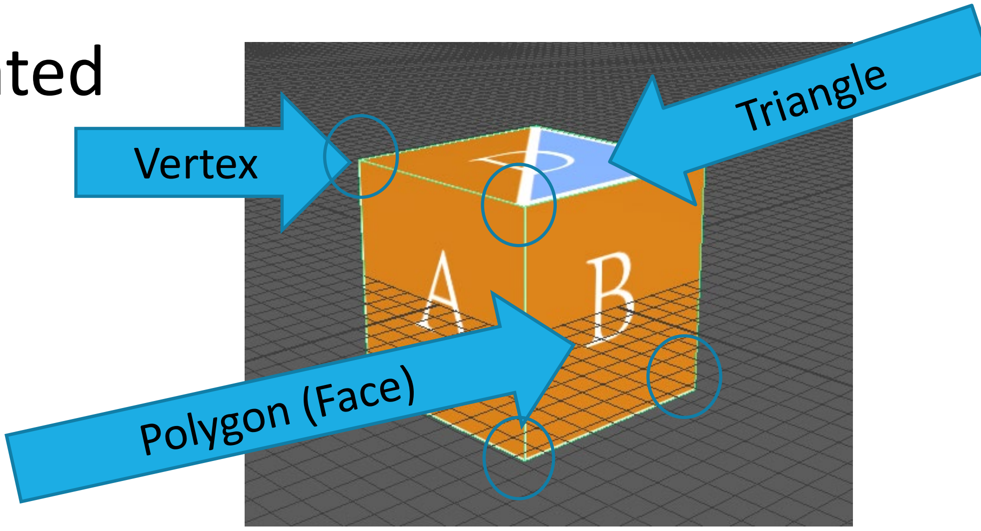




The game code sees

Mesh Recap

- Mesh
  - Vertices are (x,y,z) positions of object bounds
  - Polygons are a number of Vertices, which make a face
  - Tri's (triangles), before rendering all polygons are turned into triangles
  - Cube = 8 vertexes
    6 Faces
    12 Triangles
- The mesh can be animated
  - Skeletal animation
  - IK
  - Code



Rendering is expensive

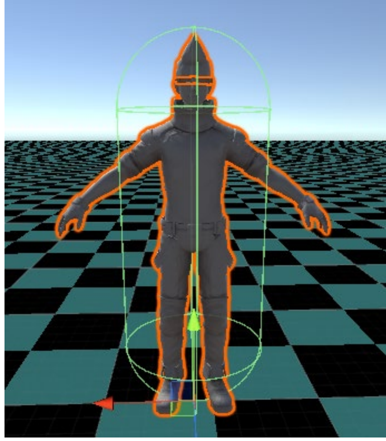
- Lots of Poly’s
- Big textures
- Dynamic lights / objects
- Transparency (read/modify/write)

- Drawing is slow so do all you can to avoid actually drawing stuff
- Lots of code is devoted to figure out what does not need drawing

Core 3rd person concept

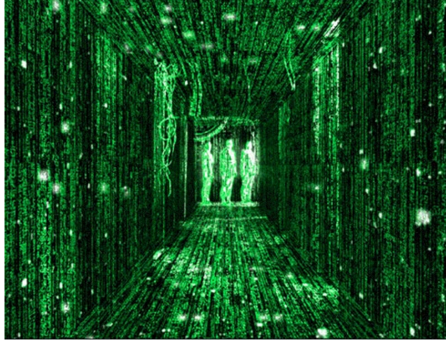
- Player moves the character
- Camera follows the player
  - Tries to avoid walls
- Unity Standard Assets
  - ThirdPersonController
  - AIThirdPersonController
  - MultipurposeCameraRig
- Lots of other files, many not needed

Separating Form & Function

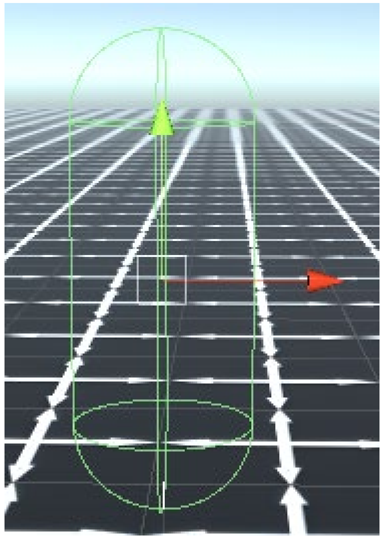


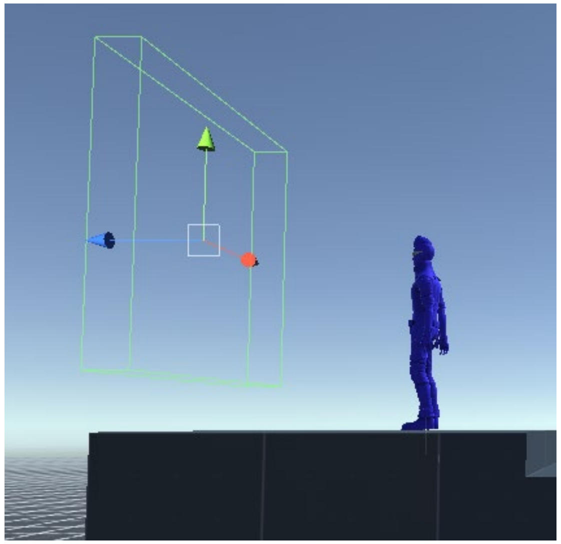
- ThirdPersonCharacter.cs
  - Drives Animation & actions collision
- AICharacterControl.cs
  - Interfaces with ThirdPersonCharacter.cs and tells it what to do for NPC's
- ThirdPersonUserControl.cs
  - Interfaces with ThirdPersonCharacter.cs and tells it what to do Player's

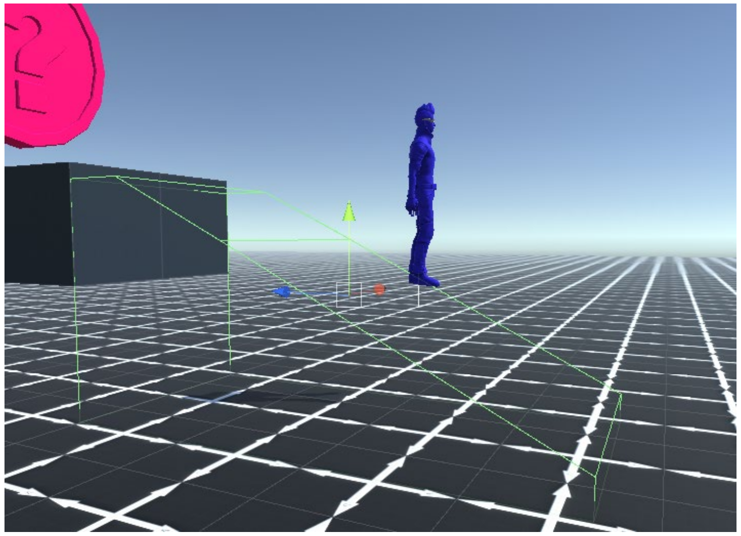
Welcome to the real (coders) world



- The game deal with simple shapes, as far as it knows Ethan is just a Capsule
- The steps are a ramp (Mesh Collider)
- The pickup a BoxCollider

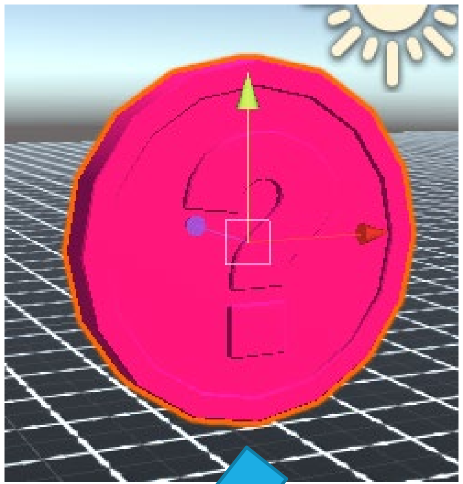


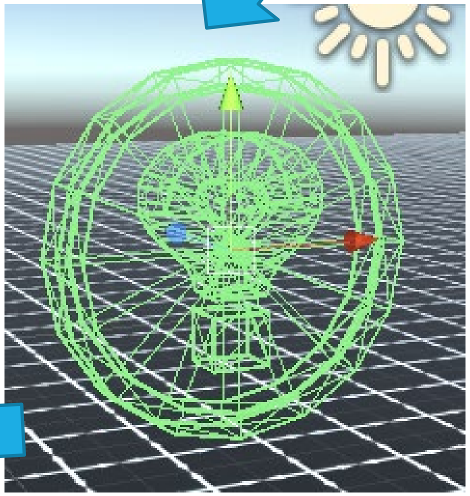


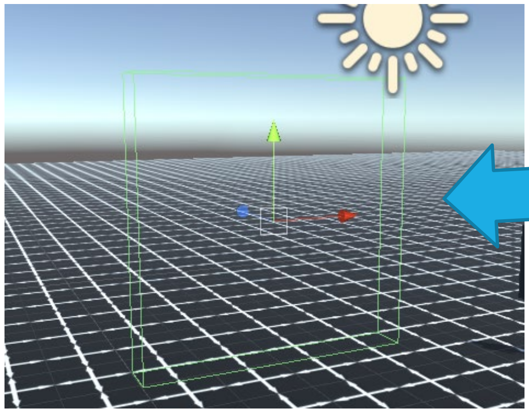


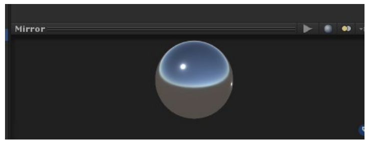
Gameplay coding

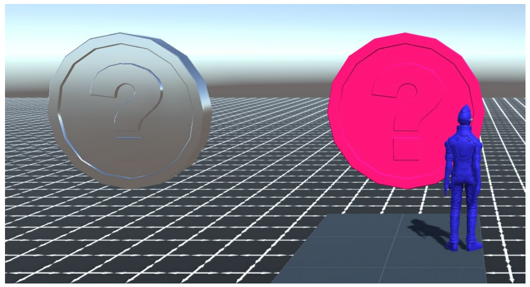
- Simplifying everything
- Maximising reuse, same object different look
- Use eye candy, tricks & cheats to hide the repetition





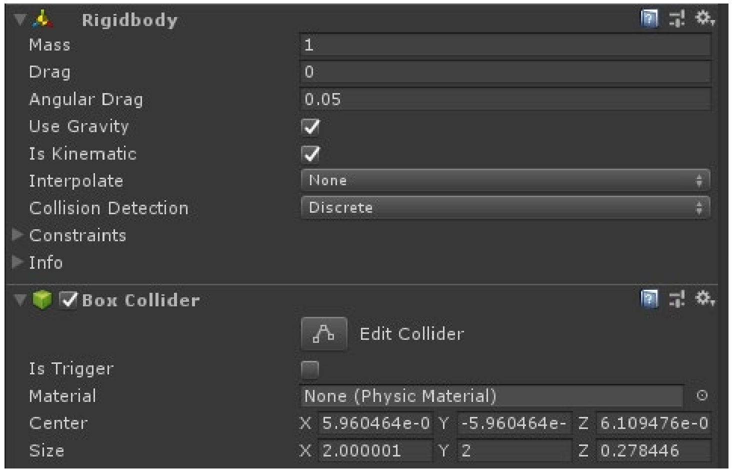






Collisions, Triggers & Physics

- Collisions are handled by physics systems
- The need a RididBody & Collider to work
- A Collider will stop another collider going through it, like the floor
- However if its set to Trigger it will only notify the script, but allow objects to penetrate
- Demo



Rigidbody

Mass	1
Drag	0
Angular Drag	0.05
Use Gravity	✓
Is Kinematic	✓
Interpolate	None
Collision Detection	Discrete
Constraints	
Info	

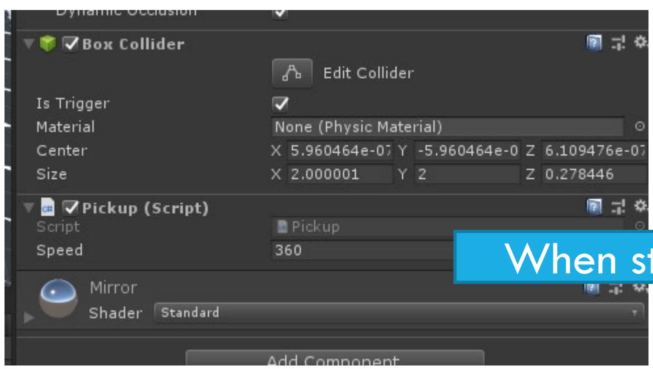
✓ Box Collider

	Edit Collider
Is Trigger	
Material	None (Physic Material)
Center	X 5.960464e-0 Y -5.960464e- Z 6.109476e-0
Size	X 2.000001 Y 2 Z 0.278446

Interaction & Animation with code

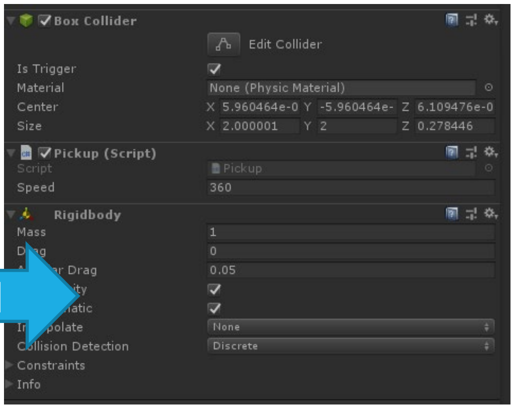
- Code can be attached to Objects to animate them
- Code can also deal with Triggers/Collisions
- We can use code to add Components, this is less error prone and requires less maintenance

```
// Add RB & set it to Kinematic
void Start()
{
    Rigidbody tRB=gameObject.AddComponent<Rigidbody>();
    Debug.Assert(tRB != null, "Cannot attach RB, may be duplicate, so remove any added in IDE");
    tRB.isKinematic = true; //Make it non physics
}
```



```
Dynamic Occlusion     ✓
Box Collider
                      Edit Collider
Is Trigger            ✓
Material              None (Physic Material)
Center                X 5.960464e-07 Y -5.960464e-0 Z 6.109476e-07
Size                  X 2.000001     Y 2             Z 0.278446
Pickup (Script)
Script                Pickup
Speed                 360
Mirror
Shader                Standard
                      Add Component
```

When started

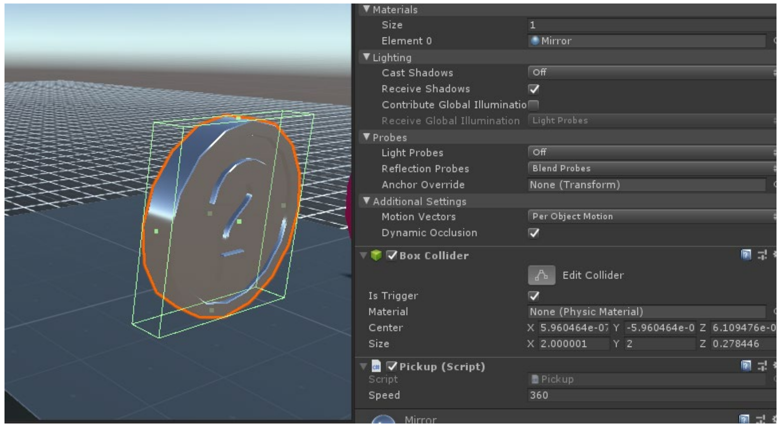


```
Box Collider
                      Edit Collider
Is Trigger            ✓
Material              None (Physic Material)
Center                X 5.960464e-0 Y -5.960464e- Z 6.109476e-0
Size                  X 2.000001    Y 2            Z 0.278446
Pickup (Script)
Script                Pickup
Speed                 360
Rigidbody
Mass                  1
D  ag                 0
A   ar Drag           0.05
      ity             ✓
      natic           ✓
In  polate            None
Collision Detection   Discrete
Constraints
Info
```

Code vs IDE

➢ Rule of thumb

1. Does the component have any elements which need to be adjusted visually, e.g. the collider?
2. Are there component variables which are helpful at runtime?
3. Are there unique, per component variables

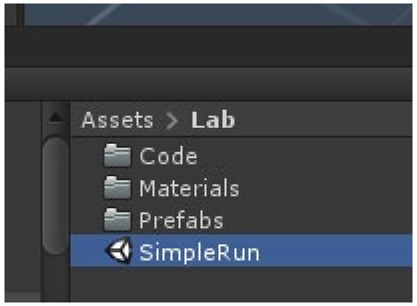


```
▼ Materials
    Size                          1
    Element 0                     Mirror
▼ Lighting
    Cast Shadows                  Off
    Receive Shadows               ✓
    Contribute Global Illuminatio ☐
    Receive Global Illumination   Light Probes
▼ Probes
    Light Probes                  Off
    Reflection Probes             Blend Probes
    Anchor Override               None (Transform)
▼ Additional Settings
    Motion Vectors                Per Object Motion
    Dynamic Occlusion             ✓
▼ ✓ Box Collider
                                  Edit Collider
  Is Trigger                      ✓
  Material                        None (Physic Material)
  Center                          X 5.960464e-07 Y -5.960464e-0 Z 6.109476e-0
  Size                            X 2.000001     Y 2            Z 0.278446
▼ ✓ Pickup (Script)
  Script                          Pickup
  Speed                           360
  Mirror
```

➢ Use IDE, if **not** code it

Workshop

Download code base



- https://github.com/RLTeachGit/Unity3rdPerson.git Run & test
- All you code should go into Lab/Code
- If you want to use assets from other folders make a duplicate and put it into the Lab Folder (use Ctrl-D to duplicate)

1. Add a Score variable to the Pickup, ensure it can be edited in the IDE (public or [SerializeField])
2. When player hits the pickup have the GM add the score using the Score setter
3. Make the pickup disappear
4. Add 5 more pickups and some platforms to player needs to navigate
5. Harder: Add a time limit, which counts down on screen