

# GET YOUR GAME ON WITH C# AND UNITY

BY JUSTIN MORGAN, JON PERKINS, AND ALEX DAUFEL

---



- Contact
  - Jon “Perk” Perkins – perk@handcannongames.com
  - Justin Morgan – justin@handcannongames.com
  - Alex Daufel – alex@handcannongames.com
- More Information about Hand Cannon Games  
<http://www.handcannongames.com/>

# UNITY – WHAT IS IT?

- What is Unity?
  - <https://unity3d.com/>
- Why use it?
- Can I get it?
- How much does it cost?



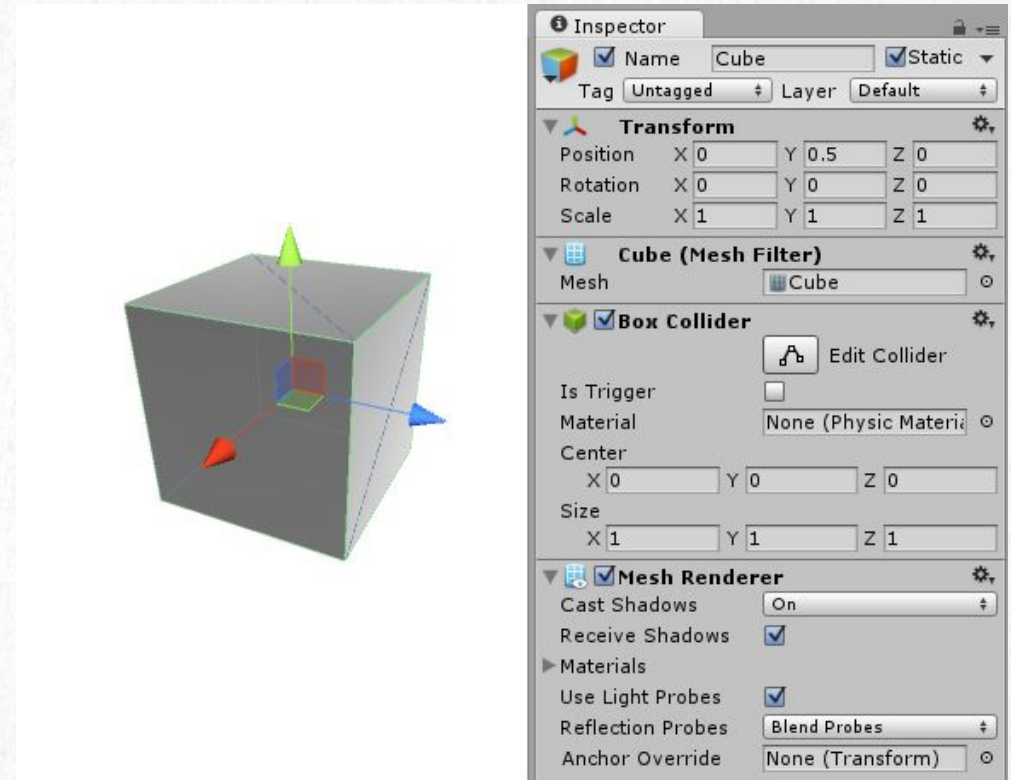
# GETTING STARTED – SCENE, LIGHTS, CAMERA, ACTION!



- Scene
- Lights
- Camera
- Assets
  - Folder Structure
  - Unity Assets Store
- Model
  - Model Files
  - Textures

# GAME OBJECT

- What is this?
- Why do I care?
- Components
  - Transform
  - MeshFilter
  - Renderer
  - Rigidbody
  - Scripts
  - Etc.



# MOVEMENT

- Create C# script
    - Start() method
    - Update() method
    - “Think in 60<sup>th</sup> of a second...” ~ Wise Game Dev
  - Get the associated Rigidbody
  - User Inputs
  - Rotation
  - Velocity and Relative Force
-

# INSTANTIATION

- Prefab
  - Library of preconfigured Game Objects
  - Template Games Objects
  - Guaranteeing exact copies of Game Objects
- Scene Controller
  - What is the point?
    - Scene transitions
    - Separation of concerns
    - Managing the chaos of Game Object interactions



# SHOOTING

- Create a Shot Prefab
  - Colliders
    - What are they?
    - Why use them?
  - Spawn Locations
  - Shot timer
    - Delta time
  - Sounds
    - Audio Source
    - Mixers
  - Destruction of Game Object
    - Memory leaks
-



## DEBUGGING

- Code Structure
  - Single Focus Methods
  - Abstract out custom code into their own methods
- Debug.log()
- Attach to Unity
- Google is your friend
- <https://docs.unity3d.com/Manual/index.html>

## TESTING

- Game testing is HARD
- Branching logic is much larger
- Testing in mind from day 1
- Test scenes
- Remove the chaos of a running game
- Still deciding good unit testing practices



# AGILE

Applying it to Game Development