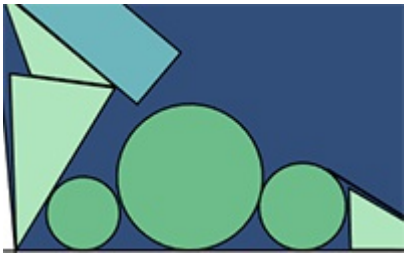
Draw 2D Physics Shapes in Unity3D

This tutorial project provides an example of how to draw shapes with the cursor in Unity similar to IncrediBots or Phun (now Algodoo).

The core logic of this project uses vertices specified from the cursor's coordinates to dynamically generate and configure a mesh, outline, and 2D collider.

- Go to the Tutorial to read about how it works in greater detail.
- Play it in your browser
- Unity Asset Store Bundle



Project Overview

Requires Unity3D (tested with 2017.x, but should also work with 5.x)

```
.
├── _Scenes
│   └── Main.unity              - The project's main scene
├── Prefab
│   ├── Circle.prefab           - Circle physics shape
│   ├── Platform.prefab         - Stationary rectangular physics platform
│   ├── Rectangle.prefab        - Rectangle physics shape
│   ├── RotatingPlatform.prefab - Rotating rectangular physics platform
│   └── Triangle.prefab         - Triangle physics shape
└── Scripts
    ├── DrawCircle.cs           - Creates circle meshes and colliders
    ├── DrawController.cs       - Captures mouse input and creates shapes
    ├── DrawRectangle.cs        - Creates rectangle meshes and colliders
    ├── DrawShape.cs            - Base class for all shapes
    ├── DrawTriangle.cs         - Creates triangle meshes and colliders
    ├── ExplosionController.cs  - Generates explosion forces at the cursor
    ├── TestPolygon.cs          - Demonstrates how to draw polygon meshes
    ├── Triangulator.cs         - Generates triangles from polygon vertices
    └── Util.cs                 - Provides handy operations on vectors
```

Running the Project

To run the project, open _Scenes/Main.unity and click on the play button.

In the main scene there will be three buttons: square, circle, and triangle. Click on one of the buttons and then click 2-3 points in the game view to draw and release a physics shape. The shape will react to gravity and interact with other objects in the scene.