GIRLS TEACHING GIRLS TO CODE: CODECAMP 2016 GRAPHICS TRACK

CREATE A GAME IN UNITY

UNITY OVERVIEW

Unity is a Game Engine, a piece of software that has both **pre-written rendering and physics code.**
People utilize it to make games!

A **Game** is a collection of **Scenes,**
which have **Objects** that are composed of **Components**.

The Objects that we are concerned with have three main Components: The Controller, the Sprite Renderer, and the Collider+Rigidbody.

The Controller tells the object how to behave,
The Renderer tells the object what it should look like,
The Collider+Rigidbody 'does physics'.

Your job is to write the controllers for our main character and our bullets, and stop the enemy invasion!

THE BASICS: CHARACTER CONTROLLER

Remember the game we showed to you in the beginning.
What does the character need to do?
Now, we want to write the controller for the character to make those things happen.

For each object in Unity, there are 2 basic methods in their controllers: **Start** and **Update**.
Start is called at the beginning of the game,
and Update is called every frame!
But besides these two methods, we also need to write additional ones for the Character. (Think about it in terms of sandwiches. Character is a PB&J, Bullets are BLTs...)

After we go through one part of the controller together, pair up with a partner and write the rest on your own!

NEXT STEP: BULLET CONTROLLER

If you successfully finished the character controls, you might notice that the bullets don't work properly. Discuss with your partner what needs to be done, then go into the bullet controller code and fix it!

TO INFINITY & BEYOND: RENDERER AND MORE!

Now you've mastered the controllers and have your game working, it's time to customize and prettify your game! There are many art resources in the Resources folder, and also feel free to add your own.

If you're feeling adventurous, considering looking at the code that's already there in Enemy Controller and Bullet Controller. Refer to the ingredients list here to see what they do, then modify that and see what happens!

Things you could play around with include:

- Change the color of the color flashes
- Make enemies faster
- Add obstacle blocks to the scene so you have obstacles that bullets bounce off of

If you have a super cool idea you want to do that is not on this list, and you don't know how to write it, let us know! We're here to help.

JUST KEEP CODING, JUST KEEP CODING

After getting this far, you can almost go home and create an entire new game on your own from scratch.

One catch though is that the ingredients we gave you were actually 'preprocessed' in order to fit everything into the short time we have. (Unity provides RawEggs(). We wrote HardBoilEggs() using RawEggs() so you can use HardBoilEggs() in MakeEggSaladSandwich().)

All of this code is in the Base folder!
For those of you who want to learn more about Unity, feel free to start in the Base folder to see what raw ingredients Unity has. They also have some great tutorials to get you started at
https://unity3d.com/learn/tutorials

Keep coding and having fun :)

INGREDIENTS FOR OUR GAME RECIPE

Here are some methods you can call, as well as descriptions of their use. This list is not exhaustive, so feel free to come up to us to learn more!

MoveTo(position);
This places our object horizontally at POSITION.
ShootBullet(bullet,velocity);
This shoots a BULLET object with a speed of VELOCITY.
GetColor();
This gets the color of our object.
SetColor(color);
This sets the color of our object to COLOR.
SetTemporaryColor(color,duration);
This sets our object's color to COLOR and changes it back after DURATION seconds.
SetSprite(sprite);
This sets the sprite of our object to SPRITE.
SetTemporarySprite(sprite,duration);
This sets the sprite of our object to SPRITE and changes it back after DURATION seconds.
LoseLife();
This deducts a life from our main character.
AddToScore(increase);
This adds INCREASE to our total score.
Remove();
This deletes our object from the scene.
RemoveAfterDelay(delay);
This deletes our objects after DELAY seconds.
Kill(collision);
This kills the enemy corresponding to the COLLISION.
SetEnemySpeedRange(min,max);
This sets the velocity range of our enemies.
Input.GetKey('key');
This is TRUE if 'KEY' is held down, FALSE if it is not.
Input.GetKeyDown('key');
This is TRUE if 'KEY' is pressed on that frame, FALSE if it is not.

And this is a cheat sheet for logical operators :) !

Operator	Meaning	Example	Value
==	equals	1 + 1 == 2	true
!=	does not equal	3.2 != 2.5	true
<	less than	10 < 5	false
>	greater than	10 > 5	true
<=	less than or equal to	126 <= 100	false
>=	greater than or equal to	5.0 >= 5.0	true