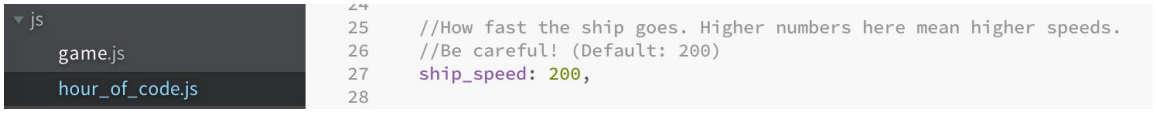
Hour of Code
Programming Challenges

In this project, you'll be changing the rules of a basic game by modifying its source code.

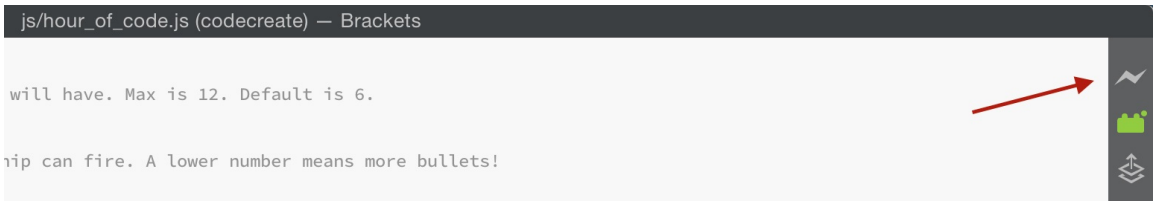
Here are some challenges to solve. You will find hints to help you along.

Part 1 - Change some settings

1. Open the program called Brackets (most likely on your desktop)
2. Open up the `codecreate` folder within Brackets (File → Open Folder). Once you have it open, you will see some files and folders. We'll be starting out with the `hour_of_code.js` file, inside the `js` folder.



3. To launch the game, click the lightning button on the top right corner in Brackets. You should now be able to play the game.



4. **Remember: lines that start with "//" (two slashes) are ignored in the program!** They are called "comments", and are left as notes for others to read.

   Now let's get back to the `hour_of_code.js` file. Notice there are some settings you can change here. For example, towards the bottom, there is a line that says:

   ```
   ship_speed: 200,
   ```

   Try changing the number to a bigger number, such as 500, and **save the file**. Now if you go back to your game, you should see that your ship goes faster.

5. If you want to add more enemies to the game, find the line that says:

   ```
   number_of_enemies: 6,
   ```

   and change it to something higher, like 12, which is the maximum.

6. Feel free to try changing some other values. For example, you can try making your spaceship bigger.

7. Now let's do something a bit more involved. Let's make the enemies shoot at random times. To do this, find the function called `enemy_fire_speed`. Inside the function, between the curly braces, you'll see a line that says:

   ```
   return 2000;
   ```

   Try changing it to

   ```
   Math.random() * 2000;
   ```

   and save the file. You should now notice the enemies firing more randomly. `Math.random()` is a function that represents a random number between zero and one, and the star `"*"` is the "times" sign.

Part 2 - Change some images

1. Let's start changing how some things look in the game, such as our spaceship. To do this, we will need to load up the file called `game.js` in Brackets.

2. Look at line 75 in `game.js`.

   ```
   // this is your spaceship
   game.load.image('spaceship', 'images/ship2.png');
   ```

   Notice that a file is being loaded as the picture for the spaceship. We can change this to a different image file.

   Try changing it to a horse:

   ```
   game.load.image('spaceship', 'images/horse.png');
   ```

   This works because there is a file called `horse.png` inside the `images` folder.

3. You may open up the `images` folder and find other pictures that you might like to use!
   You can also do the same for the enemies.

Part 3 - Change some rules!

1. Now let's make our ship invincible.

   In `game.js`, go to line 267.

```
263             enemyHitsPlayer: function(player,bullet) {
264                 bullet.kill();
265
266                 //  And create an explosion :)
267                 methods.explode(player);
268                 methods.restart(true);
269             },
```

   You can try getting rid of the lines that say:

```
methods.explode(player);
methods.restart(true);
```

   by either deleting them, or adding "`//`" at the start of the two lines.

   You should now notice that nothing happens when an enemy bullet hits your ship. This is because the code you just removed was responsible of making an explosion, and restarting the game.

2. Make the enemies explode when your ship hits them

   Find line 235, where you should see the `playerAlienCollision` method.

```
233             playerAlienCollision: function(player, alien) {
234                 alien.kill();
235                 player.kill();
236
237                 // Create some explosions :)
238                 methods.explode(alien);
239                 methods.explode(player);
240
241                 //Restart
242                 methods.restart(true);
243             },
```

   Here, there are two separate lines we'll need to remove. First is to prevent our player from dying:

```
// player.kill();
```

   And the second is to prevent the game from restarting:

```
// methods.restart(true);
```

   With these two lines disabled (or removed), now it should just be the enemy that explodes when you touch them!

Congrats! You have completed our programming challenge.
The rest is for you to explore. Keep coding!