Code Club Challenge: Space Invaders

TODO

- Images

Setting up

One of the very first computer games was called "Space Invaders"[1] and we are going to try to recreate it in Python.
To get started, open LXTerminal and use the following command to download the start of the project:

Listing 1: Get the the starting point

```
$ git clone http://github.com/jrmhaig/space_invaders
```

This should create a new directory called `space_invaders` containing a Python program and some icons. Open the program with:

Listing 2: Opening the starting point

```
$ cd space_invaders
$ idle space_invaders.py
```

Note

We need to use idle (for Python version 2) instead of idle3 (for Python version 3) because the PyGame module only works for Python version 2 on the Raspberry Pi at the moment.
You could also launch Idle from the desktop icon and then browse to the correct directory to open the file.

Controlling the ship

The first thing to do is to add the ship that you (the player) will control. This will move left and right along the bottom of the screen while the aliens, which you have to shoot, will go at the top. Look at the script below and see what you need to change in the program you have written in the first step.
Can you identify the parts that:

- Finds the picture file to use for the ship?
- Tells the computer where to put the ship?
- Finds out if you are pressing the left or right buttons?

Make the movement easier

You have probably noticed that the movement of the ship is not very easy. You really want to be able to hold the key pressed rather than having to press it lots of times. The reason it is acting the way it is is because when it detects a keyboard *event* with `event.type == pygame.KEYDOWN` it sees the single action when you press the key and moves the ship once. What we really want is for it to tell when the key is pressed and then keep moving the ship until it is released, with `event.type == pygame.KEYUP`.

[1] It was apparently originally created in 1978, which makes it almost as old as me!

Add a bit of *class*

Now we have managed to get the ship moving more easily but do you see what happens when you go to the edge of the window? We will solve this later. First, we are going to introduce a new programming idea; *Classes.*
So far, we have just a single thing in our game – a ship that you can control with the arrow keys. Later we will also have a number of aliens as well as bullets, and all of these have very similar information about them and actions. For example, they all have a position (x and y) and they all need to move. With a Class we can write code for them all once and only once.

Note

Note, some of the lines marked "Changed" have very small changes, even just a single character, while others replace several lines with just one.
Can you see how I stopped the ship going off the side of the window?
Can you work out how to make it move faster or slower?

Explanation

In the previous section we added a *Class* called `GamePiece`. By itself, this does not do anything to it provides a pattern from which we created an *Object* for the ship. Later, we will add another object for an alien and our code will contain:
Each of these is separate from the other and has its own x and y coordinates. They also both know how to move and draw themselves in the correct position in the window if we call the functions:
and we can even set their speeds with:

Note

This may be the answer to one of the questions in the last section!

As well as the `move` and `draw` functions you will see another function called `__init__` (that is a double '_' before and after the word 'init'. This is a special function that is always run once when the object is created with `ship = GamePiece(320, 410, ship_image)`. This makes sure that all the variables have the correct values at the start.

Add an alien

The real Space Invaders game has lots of aliens in rows but for the moment we will start with just one. As I said in the last section, a lot of the work is already done for us as we can use the `GamePiece` class.

Note

I am not going to write out the whole code from now, only the parts that need to change.

First, make some changes to the `GamePiece` class to let the alien "bounce" when it reaches the side of the screen.
Next, this is all that is needed to create the alien.

Shoot the alien

The next thing to do is to let you try to shoot the alien. For this, we need to load an image of a bullet with this line (try to find the correct place to put it):
Now the bullet needs to move up the screen, rather than left or right, and then disappear when it reaches the top. For this we need to add some new variables to the `__init__` function:
and add a new part to the `move` function:
At the start of the game, the bullet doesn't exist and we can indicate this with:
And finally, the bullet should appear when we press the 'SPACE' bar.

Hit the alien

You may have noticed that the bullet currently just goes straight through the alien. This isn't very good! In the `GamePiece` class we need a new function to detect if it has been hit.
At the moment there is only one alien so when it is hit we will put it back at the top and make it move faster. Later, we can have several rows of aliens and try to keep a score.

More aliens

TODO
and lose if aliens reach the bottom.

Keep score

TODO

Aliens' bullets

TODO

Next levels

TODO