SAMPLE WORKSHEET

In this task we will be coding a Flappy Bird style game using JavaScript's 'Phaser' library.

STEPS:
(1) Contain our game in a Phaser.Game subclass and code the game contents within a Phaser.State subclass.
(2) Preload all of the assets.
(3) Create our bird and program its associated behaviour.
(4) Code the pipe columns that our bird will navigate through.
(5) Add some basic animation and sound to the bird.

---

(1)

This basic game will contain two classes - one that extends the `Phaser.Game` class, essentially acting as the container for our game to run in, and one that extends the `Phaser.State` class, which contains the working code for the game.
Remember that given that we are extending these existing classes, you need to run `super(…)` in the class `constructor(){…}` method to set parameters that you would normally set when instantiating a `Phaser.Game` object.
Create `Game`, a class extending `Phaser.Game`. Given that our whole game will run by instantiating a `new Game()` object, ideally which class method should all of the working code `Game` sit so that it executes?

Build the constructor of the `Game` class, and use `super(…)` to:
- Set the dimensions of the game. We recommend 400x490 pixels.
- Set the mode to `Phaser.AUTO`.

Create `Main`, a class extending `Phaser.State`. Within it, you should create blank `preload`, `create()`, and `update` class methods which are required for `Phaser.State` object behaviour.

---

(2)

Type the following into `preload` to configure the display:

```
// DISPLAY CONFIG
if(!this.game.device.desktop) {
    this.game.scale.scaleMode = Phaser.ScaleManager.SHOW_ALL;
    this.game.scale.setMinMax(this.game.width/2,
    this.game.height/2, this.game.width, this.game.height);
}
```

You'll notice that `this.game` is used as a reference to the currently running `Phaser.Game` instance in, which will correspond to our `Game` class. Using methods of `this.game` in a similar fashion:
- Set the background colour of the `stage`.
- From our assets folder, `load` in our bird and pipe images, and our jump audio (the sound when the bird flaps upwards).