Phaser Quick Guide

Boilerplate/Template

```
function preload(){
// Load in assets from paths (local or external)
}

function create(){
// Contains code to render GameObjects
}

function update(){
// Executes code once every animation frame
}

const config = {
// Contains properties with metainformation
};

// Creates and starts the Phaser game
const game = new Phaser.Game(config);
```

Legend

`preload()`

A function in the Scene object that loads in assets like images and sounds.

`create()`

A function in the Scene object that creates the game's sprites, images, Colliders, Groups, and so much more.

`update()`

A function in the Scene object that executes every animation frame. It can be used to check for keyboard input or manipulating GameObjects, among other things.

`config`

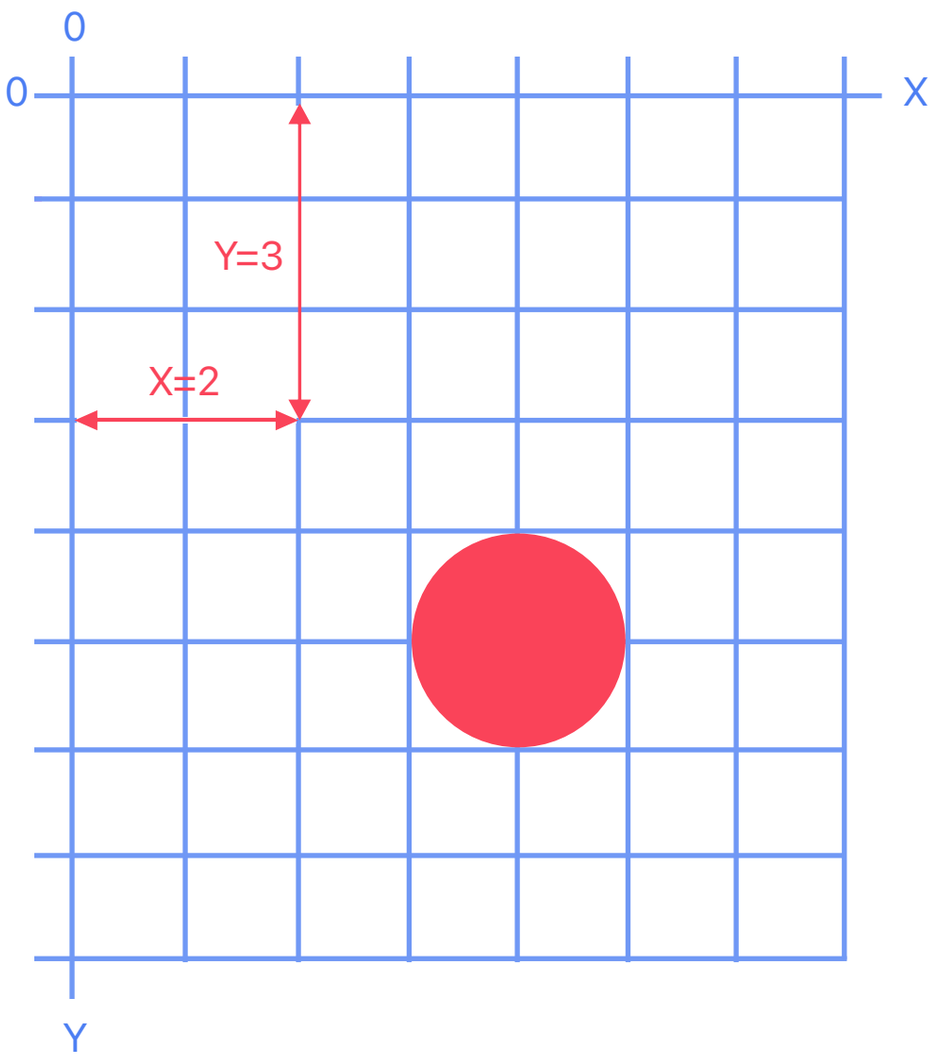
An object that contains the game's metainformation such as the height and width of the game screen, background color, the Scene's functions, and more.

`new Phaser.game(config)`

Creates a `<canvas>` element to allow Phaser to make GameObjects, starts the scene function flow, and maintains key Phaser features. The data from `config` is used to determine game settings.

Scene Function Flow

Phaser's Grid System



Notes:

The game screen can be thought of as a coordinate grid.

The origin (0, 0) starts at the top left corner.

Going from the left side of the screen to the right side of the screen increase the x-coordinate.

Going from the top of the screen to the bottom of the screen increases the y-coordinate

The coordinates (2, 3) is located 2 pixels to the right and 3 pixels down from the origin.

The circle provided can be made by:

```
function create() {
    const redHexCode = 0x7ED321;
    this.add.circle(4, 5, 1, redHexCode);
}
```

The circle's centered at the coordinates (4, 5) has a radius of 1.

The last argument provides the color.