

New World

What you need

You should have `you-win` running, with:

- a text editor, with a new game file open (e.g. a file called `game.js`).
- a Chrome window open (probably on `http://localhost:8000/`)
- (optional) a phone running Chrome or Safari, with your computer opened.

Put your text editor and Chrome window side-by-side, if you can (and/or keep your phone open in front of you!). Whenever save in your text editor, `you-win` will automatically refresh the page.

Make sure you have line numbers turned on in your text editor.

Your Chrome window should have a white screen. This is a blank canvas in which we can start making mobile games!

The template

Have a look at the template that's open in your text editor. It should have the following parts:

- These lines load `you-win`, the library we're using to help make a game.

```
const uw = require('you-win')
const {Phone, World, Sprite, Text, Polygon} = uw
```

- We then load in all the sounds and images we need. By default we'll just get some emoji.

```
// Load everything we need
await uw.begin()
```

- We then make our world, which represents the screen.

```
// Make the world
var world = new World
world.title = ''
world.background = 'white'

// Now we can start making Sprites!
```

Change the background colour

Let's change the background colour

You can use any of the colour names which are supported by HTML. If you give a colour name that it doesn't recognise, you'll probably just get black.

If you want other colours, instead of named colours you can use colours like '#007de0'. There are called a "hex code", short for hexadecimal. You can choose your own hex code colour with an online [colour picker](#).

Change it to your favourite colour.

```
label.fill = '#007de0' // whatever you fancy really
```

Set the shape of the screen

We can set the width and height of the world to change the size of the screen. (It's like the *Stage* in Scratch.)

```
world.width = 300
world.height = 460
```

Add this at the bottom of your program.

To change the attributes of an object in JavaScript, we write the name of the object, followed by a dot, and then =, and then the new value.

Save, and your world should change shape. It's now a white box, roughly the shape of a mobile phone in portrait, with black bars around the edges.

Sprites

First, we need to know how to add things to the world, so they appear on the screen.

In JavaScript, you create new things using the `new` keyword, followed by the kind of thing you want to make (such as `World`, `Sprite`, or `Text`).

We can add images to the `World` by creating `Sprites`. A `Sprite` is an image on the screen, which we can move about, rotate, flip, scale, make transparent, and so on. Let's make our first sprite.

- Add this code, after the block which makes the `World`.

```
var poop = new Sprite
poop.costume = '🐛'
```

Save. Have a look at Chrome—now there should be a tiny poop in the middle of the screen!

We use the `var` keyword so we have a name to refer to our `Sprite` with.

As before, we can set object attributes using “dot notation”, and giving it the new value.

Here are some kinds of values: *(These are just examples, don't type them in!)*

- **Boolean:** either `true` or `false`.

Example: the `poop.flipped` attribute is a Boolean. When set to `true`, it makes the Sprite face the other way.

- **Number:** e.g. `123` or `3.14`.
- **String:** some text. Strings are written with quotes around them, e.g. `'birb'` or `"potato"`. You can use either single or double quotes.

Now try this.

- **Challenge:** Set the `scale` attribute of your sprite to make it twice as big. (The scale is a number, starting at `1.0`).

Remember that you have to include the name of your Sprite, so it should be `poop.scale` rather than `scale`. It won't work to write `scale` by itself, since the computer won't know which sprite you're talking about.

When we make a sprite, we get to specify its initial values. We can change any of them later, too (more on that in the next chapter).

Here are some other properties you can try:

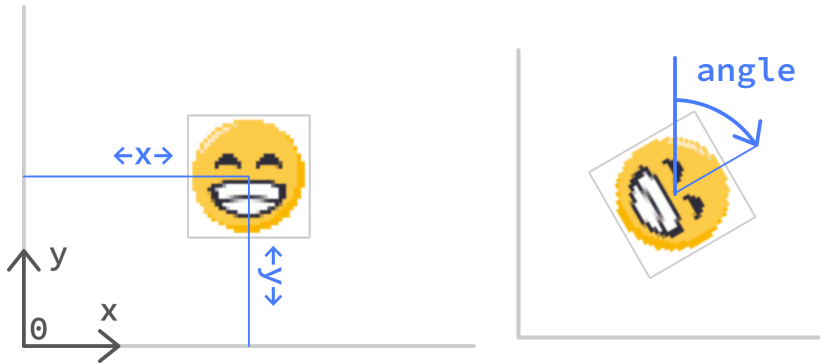
- `opacity` (a number between 0 and 1, starting at `1.0`)
- `angle` (a number, in degrees, starting at `0`)
- `flipped` (a boolean, initially `false`)

Coordinates

Let's move our sprite about. We can do this using the attributes `poop.posX` and `poop.posY`.

These are the co-ordinates of the center of the sprite, starting from the bottom-left corner of the screen.

Here's a quick diagram introducing coordinates:



- Move your poop to the position (100, 200).

```
poop.posX = 100
poop.posY = 200
```

We can add other Sprites, too!

- Add a second sprite, called cow.

```
var cow = new Sprite
cow.costume = 🐮
```

Edges

We just used the `.posX` and `.posY` attributes to set the **center** of the sprite.

We can also set the **edges** of the sprite, using the attributes `.top`, `.bottom`, `.left`, and `.right`.

- Move the cow to touch the left side of the screen.

```
cow.left = 0
```

- Move the poop to be to the right of the cow.

```
poop.posY = 100
```

Delete this line.

```
poop.left = cow.right
```

*Add this line. Make sure it's after **both** of the poop and cow variables have been created.*

Notice that you can't use the name of a Sprite before you create it. If you write `cow` in your program above the `var cow = new Sprite` line, it won't work.

- Now move the cow.

```
cow.left = 200
```

Add this line at the bottom of your program.

Notice that the poop doesn't move, even though we've moved the cow. This is important: when you set an attribute using `=`, it only happens once.

Randomness

Let's introduce some uncertainty into our creation.

We can use `uw.randomInt(1, 10)` to pick a random number between 1 and 10. (This is just like `pick random _ to _` from Scratch.)

- Move your poop to a random `posX` position.

```
poop.posX = uw.randomInt(0, world.width)
```

- **Challenge:** Move your poop to a random `posY` position.

Refresh the page. Every time you refresh, the position of the poop should change!

Now try copy/pasting the code for the poop, to make a couple more random ones.

- Make two more poops.

Text

We've seen how to do images; now let's add some text to the screen.

When someone starts programming, it's traditional to for them to introduce themselves by saying "Hello world!". Let's do that now.

- Create a Text object.

```
var label = new Text  
label.text = "Hello world!"
```

- Change it to red.

```
label.fill = 'red'
```

The End

Good job! Now you know how to:

- Make JavaScript objects with the **new keyword**
- Use **var to name them**, so you can refer to them later
- **Set their attributes**, using dot notation and =
- Create Sprites and Text
- Set the **position** of objects inside the world
- How to **pick random numbers** using `uw.randomInt`

Let's continue on to [chapter two](#)!