
Install Chrome

You need Chrome or Chromium running on a desktop.

- **On Windows:** If you don't already have it, download from [Google](#)
- **On Ubuntu:**

```
apt install chromium-browser
```

You need Chrome running on an Android. (If you want to debug on a phone - not essential)

- Either install Chrome from [APK Mirror](#);
- or install Chrome from the Play Store, which requires a Google account.

Option #1: Glitch

You can use [Glitch](#) to code a mobile game, without installing anything.

Open our [you-win template](#), **remix** it, and start coding immediately!

Option #2: Node.js

If you prefer, you can **install** `you-win` to your computer.

1 On Windows

- Install [Node.js](#), which includes NPM.

On Ubuntu, run the following in a Terminal:

```
curl -sL https://deb.nodesource.com/setup_8.x | sudo -E bash -  
sudo apt-get install -y nodejs  
sudo apt-get install -y build-essential
```

2 Install `you-win` globally.

Run the following in your shell (**Command Prompt** on Windows, **Terminal** on macOS/Linux):

```
npm install -g you-win
```

3 Start a new project or run your existing one!

In your shell, use the `cd` command to move into your Documents folder (or whichever directory your files are already in).

```
cd Documents
```

Run the `you-win` command to make a new project:

```
you-win first.js
```

If you are trying to start an existing project, replace `first.js` with your filename (maybe it is `game.js`). You can now skip directly to the section below that says “To run your game”.

If you’re starting a new project, this will create a new game from the `you-win` template, which should save you a bit of typing.

Now open the `first.js` file you just created in your favourite text editor. Text editors include **Gedit** (Ubuntu only), **VSCode**, **Brackets**, or **Sublime Text** (search these online to find them, at Livewires we use VSCode).

The template should look [like this](#):

```
import UW from 'you-win'
import {forever, Phone, World, Sprite, Text, Polygon, Rect, Sound}
from 'you-win'

// var phone = new Phone

UW.init({
  })
.then(() => {

  var world = new World
  world.title = ''
  world.background = 'white'

  // ...

})
```

To run your game:

- Open <http://localhost:8000> in Chrome on your computer.
- In your Terminal, you should see a web address output to the console, eg `http://192.168.1.10:8000/`. Open that address on your Android. If you don't want or need remote debugging you can open it on any phone that's on your network (wifi) to see what it looks like.

Problems?

If you were at LiveWires and are having trouble getting your program to run, come find us on the [forums](#) and we'll try and help you.

Remote Debugging

ADB Driver

First, make sure you have the Android ADB driver installed.

- **On Ubuntu:**

```
apt install android-tools-adb
```

Make sure you're in the `plugdev` group.

```
sudo usermod -aG plugdev $LOGNAME
```

- **On Windows:** [Install the OEM driver](#) from your manufacturer.
- **On macOS:** “It just works. Skip this step.”

USB Debugging

You need an Android running 4.2 or later (e.g. a Galaxy S3 or newer).

On your Android phone:

- 1 In **Settings**, find **About phone**, and tap **Build number** 7 times. This should [enable Developer Options](#).
- 2 Go back, and open **Developer Options**. Enable **USB Debugging**.
- 3 Use a micro USB cable to connect your Android to your computer. Make sure to **Allow USB Debugging** on your phone.

On your desktop:

- 1 In Chrome, [open DevTools](#) (Ctrl+Shift+J; or ⌘+⌥+J on Mac). Go to **More Tools** → **Remote devices**. Make sure **Discover USB Devices** is ticked.
- 2 Find the `you-win` tab running on your phone, and click **Inspect**.

Now you can see a preview of your phone on the computer—but more importantly, you can see any error messages which appear!

Read Google's [Remote Debugging article](#) for more details.

