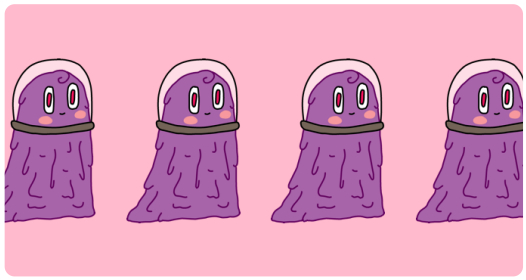
Projects

# Clone wars

Create a game in which you have to save the Earth from space monsters

Scratch



## Step 1 Introduction

In this project you will learn how to create a game in which you have to save the Earth from space monsters.

### What you will make

Score as many points as you can by shooting flying space-hippos. If you get hit by a hippo or by an orange dropped by the bats, you lose a life.


### What you will need

**Hardware**

- A computer capable of running Scratch 2.0

**Software**

- Scratch 2.0 **offline** (https://rpf.io/scratchoff)

**Downloads**

**Find the downloads here** (http://rpf.io/p/en/clone-wars-scratch2-go).

**What you will learn**

- How to make sprites move using keyboard input
- How to clone sprites to make copies of them
- How to use 'broadcast' and 'receive blocks' to send messages

**Additional notes for educators**

**If you need the solution to this project, you can find it here** (http://rpf.io/p/en/clone-wars-scratch2-get).

## Step 2 Make a spaceship

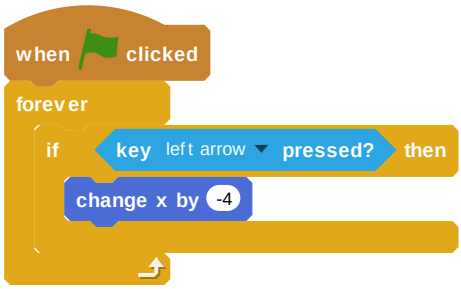
First make a spaceship that can defend the Earth!

Open the 'Clone wars' Scratch starter project.

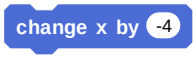
Download the starter project from **rpf.io/p/en/clone-wars-scratch2-go** (http://rpf.io/p/en/clone-wars-scratch2-go), and then open it using the offline editor.

If you need to download and install the Scratch offline editor, you can find it at **rpf.io/scratchoff** (https://rpf.io/scratchoff).

Add this code to the spaceship sprite to make the spaceship move left if the `left` arrow is pressed:

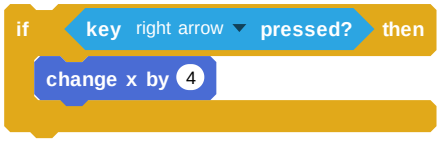



The x-axis goes from the left side of the Stage to the right side. This means that the spaceship moves to the left when you subtract from the value of the spaceship sprite's `x` position. So this code block is the part that makes your spaceship move left:




Add some more code inside the `forever` block to make your spaceship move to the right if the `right` arrow key is pressed.

Here is the code you need to add below the other code inside the `forever` block:




Test your project by clicking the green flag. Can you press the arrow keys to make your spaceship move left and right?

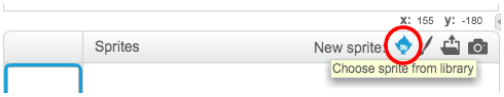
## Step 3 Lightning bolts

Now you are going to give the spaceship the ability to fire lightning bolts!

Add the `Lightning` sprite from the Scratch library.

### Adding a Scratch sprite from the Library

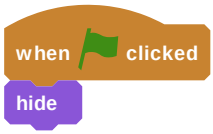
- Click **Choose sprite from library** to see the library of all Scratch sprites.




- You can browse sprites by category, theme, or type. Click on a sprite and click **OK** to add it to your project.

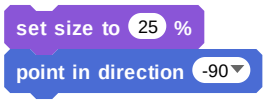

When the game starts, the `Lightning` sprite should be hidden until the spaceship fires its laser cannons.

Add this code to the `Lightning` sprite:




At the moment, the lightning bolt is really big compared to the spaceship!

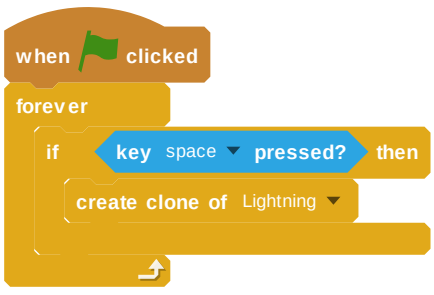
Below the code that the `Lightning` sprite already has, add some blocks to make the sprite smaller and to turn it upside down.




Now it looks like it fires pointy end–first out of the spaceship.

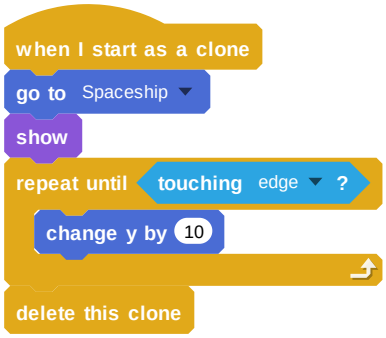
Add some new code to the `Spaceship` sprite to create a new clone of the lightning bolt if the `space` key is pressed.

Here is what your new code should look like:



Whenever the game creates a `Lightning` sprite clone, the clone should appear and then move upwards until it reaches the top of the Stage. Then the clone should disappear.

Add this code to the `Lightning` sprite so that clones of it move upwards until they touch the edge of the Stage, and then they get deleted.




Press the space key to test whether the lightning bolt moves correctly.

**Challenge!**

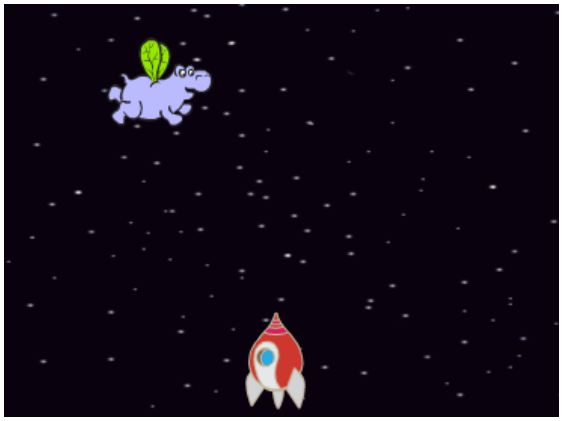
**Challenge: improve the lightning**

What happens if you hold down the space key? Can you use a `wait` block to fix this?

## Step 4 Space-hippos

Now you're going to add lots of flying hippos that try to destroy your spaceship.

Create a new sprite with the 'Hippo1' image in the Scratch library. Use the **shrink** tool to make the `Hippo` sprite a similar size to the `Spaceship` sprite.



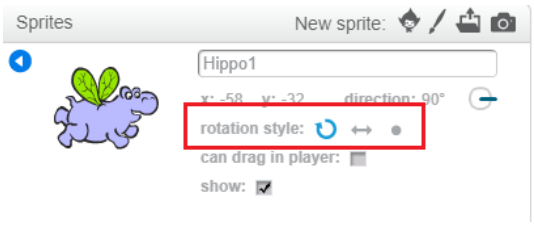
Set the `Hippo` sprite's rotation style to **left-right**.

### Set sprite rotation style

You can set which way a sprite rotates.

- Click on the blue **i** near the sprite in the **Sprites** panel.


- Click on the rotation style you want.

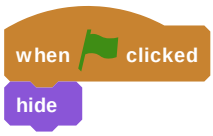



The styles are:

- Full rotation — points the sprite in the direction it is facing
- Left–right — flips the sprite left or right only

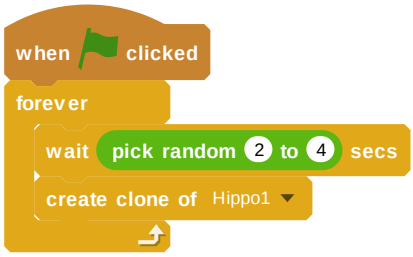
- Don't rotate — the sprite looks the same regardless of which direction it is facing

Add some code to hide the `Hippo` sprite when the game starts.




Add some code to the Stage to create a new `Hippo` clone every few seconds.

This is what your code should look like:

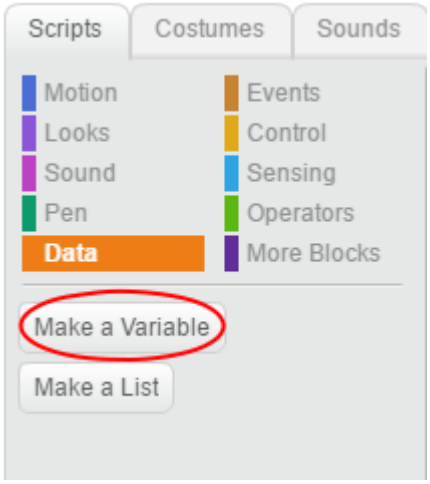



Each new hippo clone should appear at a random `x` position, and every clone should have a random speed.

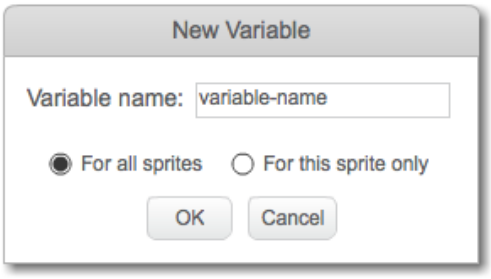
Create a new variable called `speed` that is for the `Hippo` sprite only.

**Add a variable in Scratch**

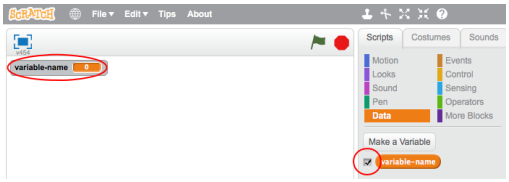
- Click on **Data** in the Scripts tab, then click on **Make a Variable**.




- Type in the name of your variable. You can choose whether you would like your variable to be available to all sprites, or to only this sprite. Press **OK**.



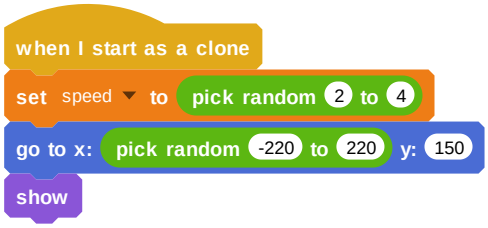

- Once you have created the variable, it will be displayed on the Stage, or you can untick the variable in the Scripts tab to hide it.




- New blocks will appear and allow you to change the value of the variable.

When you've done this correctly, the variable has the name of the sprite next to it, like this:

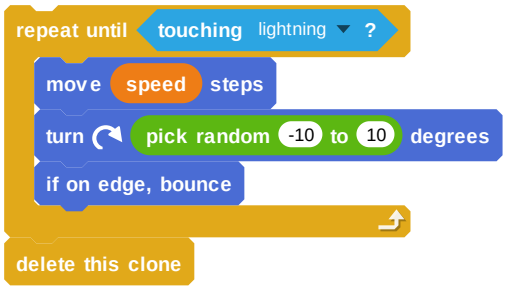

When each `Hippo` clone starts, pick a random speed and starting place for it. Then show the clone on the screen.



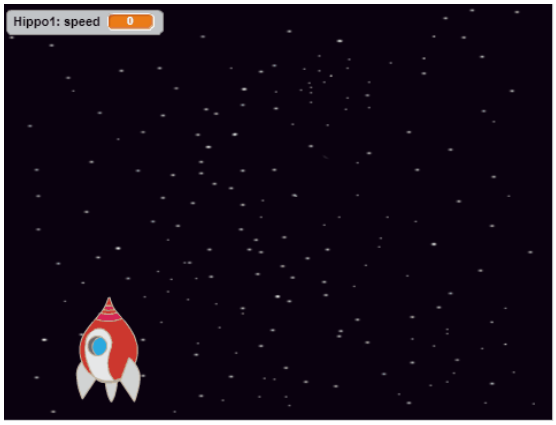

Test your code. Does a new hippo appear every few seconds?

At the moment the hippos don't move.

Each hippo should move around randomly until it gets hit by a lightning bolt. To make that happen, attach this code below the blocks that are already in the `Hippo` sprite's code script:




Test your code again. You should see a new hippo clone appear every few seconds, and each clone should move at a different speed.



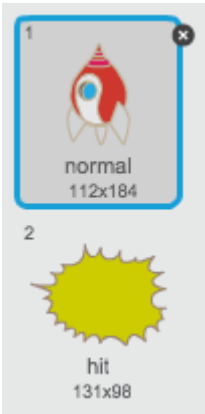

Now test the spaceship's laser cannon. If a lightning bolt hits a hippo, does the hippo vanish?

## Step 5 Spaceship explosion

When a hippo touches your spaceship, the spaceship should explode!

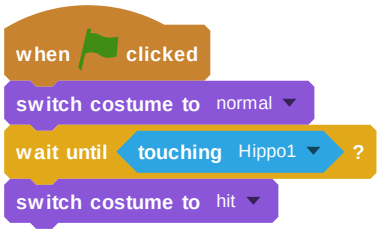
Select the `Spaceship` sprite and rename its costume to 'normal'.

Draw another costume of an exploding spaceship, and call the new costume 'hit'.




If you don't want to draw the explosion, you can select the 'Sun' costume from the Scratch library, and then use the **Color a shape** tool to change the costume's colour and face.

Add some code to your `Spaceship` sprite so that it displays the 'normal' costume when the game starts, and switches to the 'hit' costume when it touches a hippo:

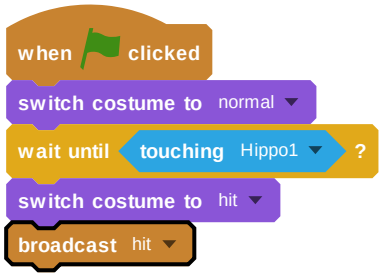



Test your code. Make the spaceship collide with a hippo. Does the spaceship change to the 'hit' costume?

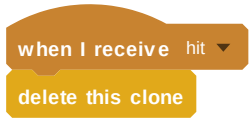
## Step 6 Hippos that disappear

When the spaceship explodes, all the hippos should disappear so that players of the game can recover.

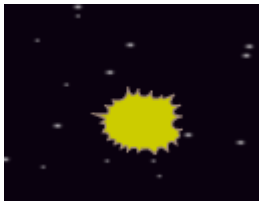
Add code to the spaceship sprite to make it `broadcast` the message "hit" when the `spaceship touches a hippo`.




All of the `Hippo` sprite clones will receive the "hit" message, and you can instruct them to disappear when the spaceship is hit by adding this code to the `Hippo` sprite:

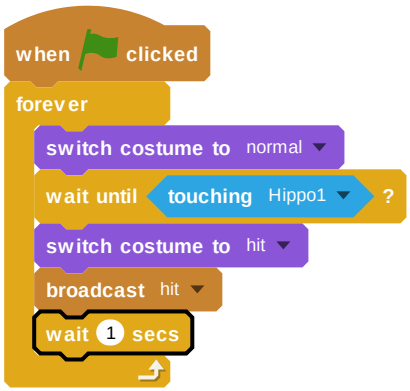



To check whether the new code works, click the green flag and make the spaceship collide with a hippo.



After the spaceship explodes, new `Hippo` clones appear, but the spaceship is still exploded! The spaceship needs to reset itself after being hit.

Add a `wait` block at the end of the `Spaceship` sprite's code to create a small pause before hippos begin appearing again.
Then add a `forever` block around all of your code to make the code run repeatedly.




**Challenge!**

### Challenge: lives and score

At the moment, you can play the game forever, but it doesn't count how many hippos you shoot or how many times your spaceship explodes.

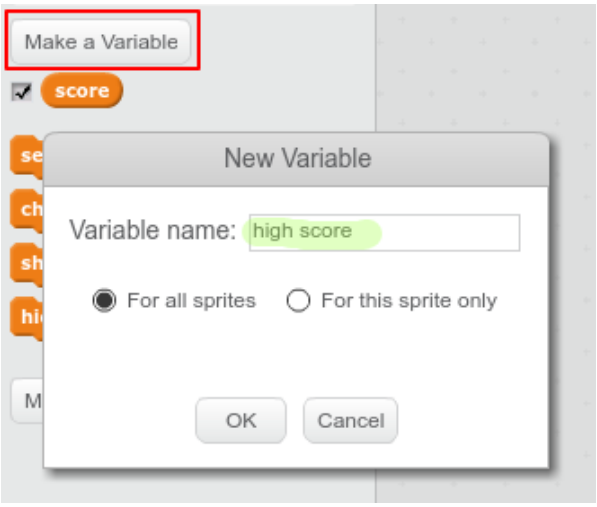
Can you add `lives`, a `score`, or even a `highscore` to your game?

#### Create a high score

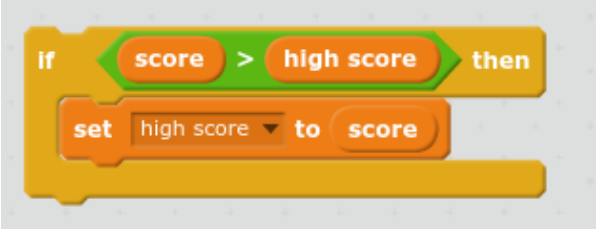
It's fun to keep track of a high score in a game.

Let's say you have a variable called `score`, which gets set to zero at the beginning of each game.

Add another variable called `high score`.

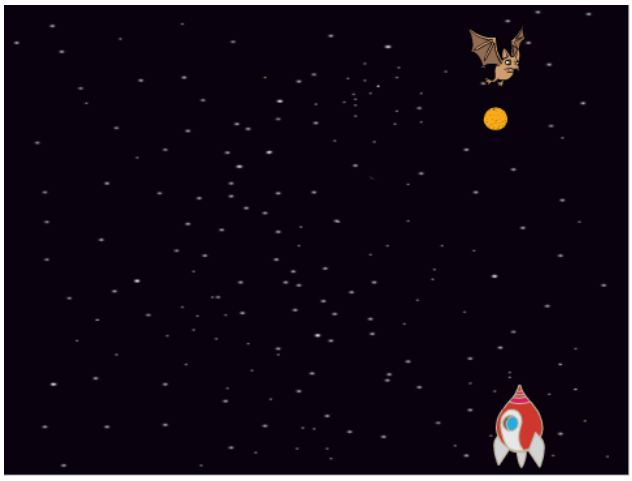



At the end of the game (or whenever you want to update the high score), you'll need to check whether you have a new `high score`.



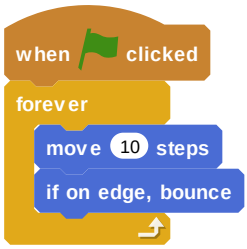

## Step 7 Space-bat

To make your game a bit harder, you are going to create a bat that throws oranges at the spaceship.



Add a `Bat` sprite and set its rotation style to **left-right**.

Make the `Bat` sprite `move` from left to right at the top of the Stage `forever`.




Remember to test your code.

If you look at the bat's costumes, you can see that it has two different ones:


Use the `next costume` block to make the bat flap its wings as it moves.

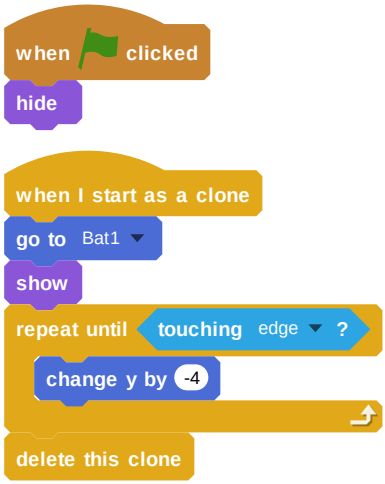
You code should look like this:


Now make the bat throw oranges!

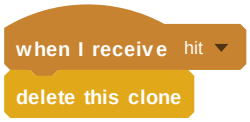
Add an `Orange` sprite from the Scratch library.


Add code to your bat so that `when the flag is clicked`, the `Bat` sprite `forever` `waits` for a `random` length of time between `5 to 10` seconds and then `creates a clone` of the `Orange` sprite.


Add code to the `Orange` to make each of its clone drop, starting from the `Bat` sprite and falling towards the bottom of the Stage.




Add some more code to the `Orange` sprite so that when an `Orange` clone hits the `Spaceship` sprite, the clone also disappears to give the player a chance to reset:




Modify the code of your `Spaceship` sprite so that the sprite is "hit" when it touches a `Hippo` sprite or an `Orange` sprite:


Test your game. What happens if the spaceship gets hit by a falling orange?

## Step 8 Game over

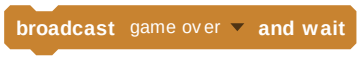
Next, you're going to add a 'game over' message at the end of the game.

If you haven't already, create a new variable called `lives`.

Your spaceship should start with three lives and lose a life whenever it touches a hippo or an orange. Your game should stop when the `lives` run out.

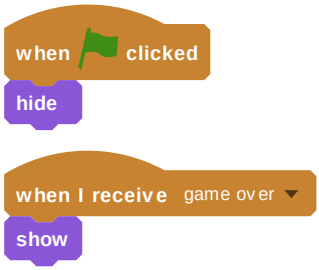
Draw a new sprite called `Game Over` using the **text** tool.


On the Stage, broadcast a `game over` message just before the game ends.




Add this code to your `Game Over` sprite so that it shows at the end of the game:

GAME OVER!




Because you've used a `broadcast [game over] and wait` block on your Stage, the Stage will wait for the `Game Over` sprite to be displayed before ending the game.

Test your game. How many points can you score? If the game is too easy or too hard, can you think of ways to improve it?

**Challenge!**

**Challenge: improve your game**

What improvements can you make to your game?

Here are some ideas:

- Add health packs that you can collect to gain extra lives.

- Add floating rocks that your spaceship must avoid.

- Make more enemies appear when your score gets to `100`.

## Step 9 What next?

Have a go at our **Create your own world** (https://projects.raspberrypi.org/en/projects/create-your-own-world) project, where you'll create your own adventure game!