

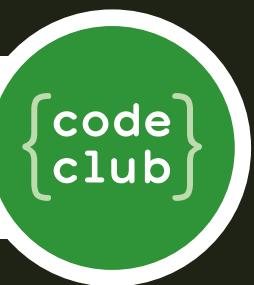


# [CHAPTER FIVE]

# BOAT RACE

Create your own boat race game,  
complete with mouse control, collision  
detection, and on-screen timer

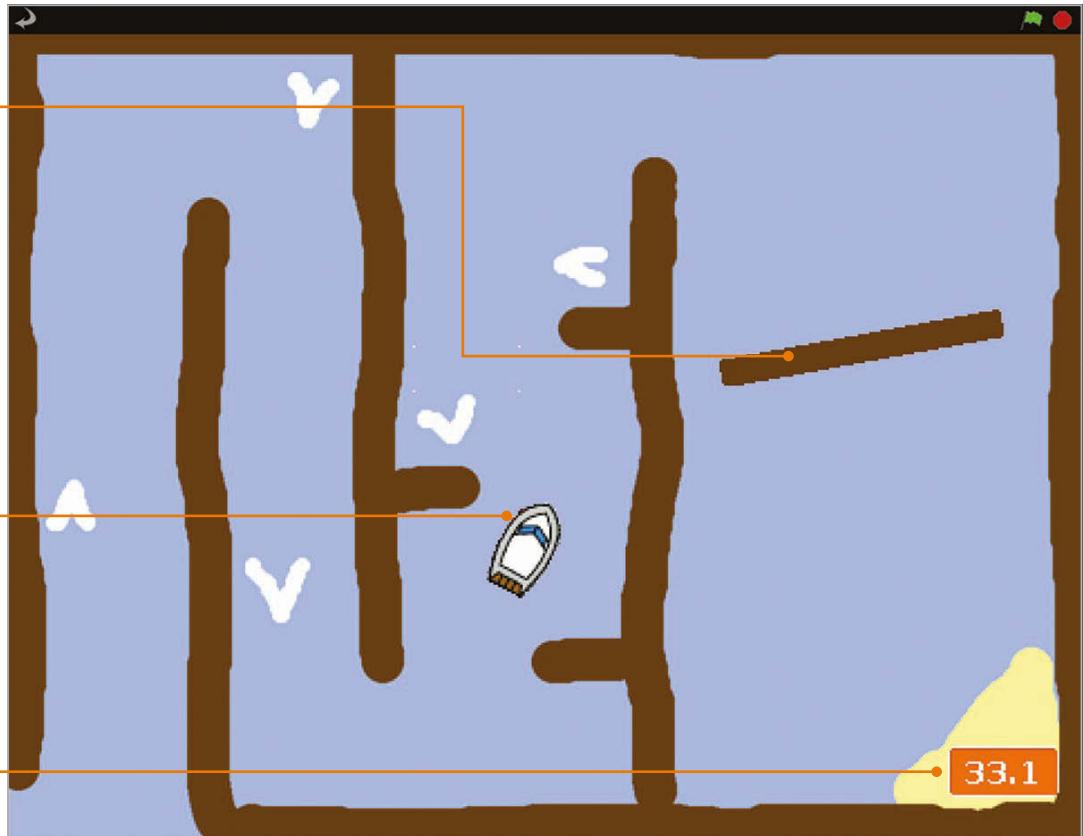
Join your local **Code Club**  
for more amazing resources  
like this: [codeclub.org.uk](http://codeclub.org.uk)



The boat crashes if it hits something brown like this revolving gate

The boat sprite is programmed to move towards the mouse pointer

The timer is shown on screen, and stops when the boat reaches the yellow beach



In this chapter, you'll be making your own arcade game in which the player attempts to guide a boat safely around a maze-like course – including a revolving gate – to the finish in as fast a time as possible. You can even design your own custom course if you like. As well as moving a sprite towards the mouse pointer, this project involves collision detection, using the **touching color** Sensing block to determine whether the boat has hit something. Let's dive in and start coding...

## >STEP-01

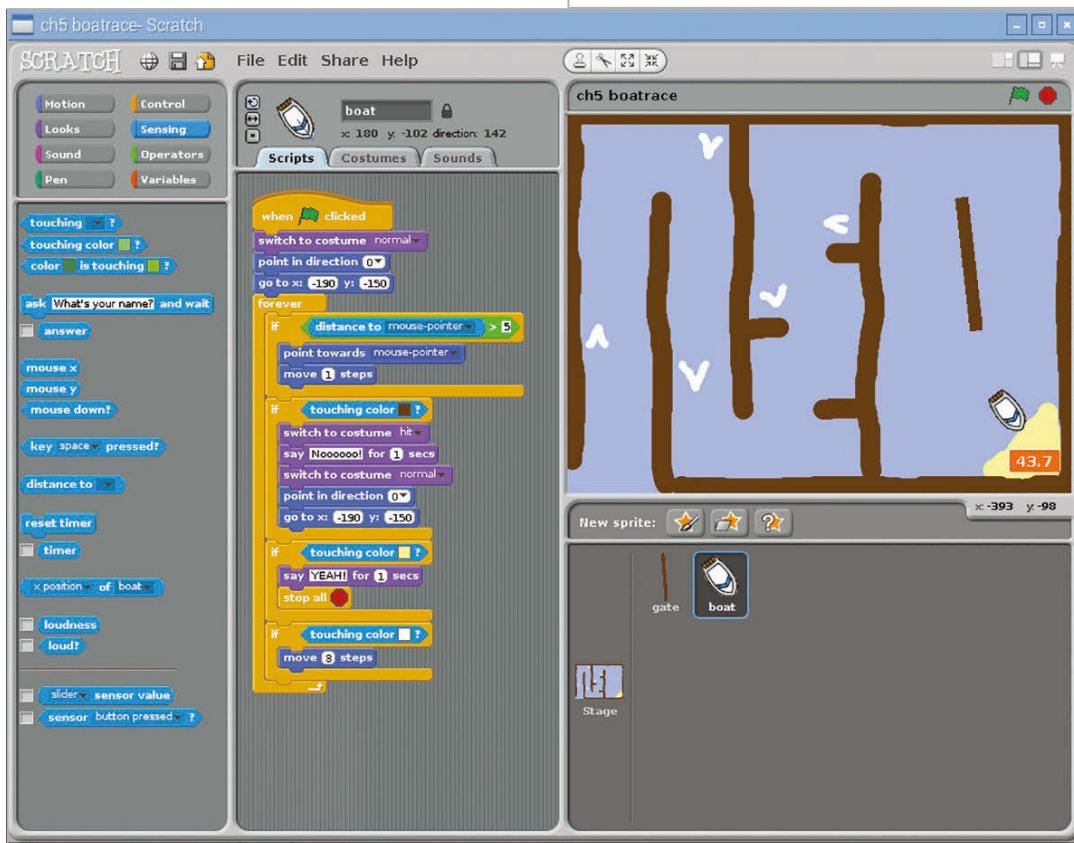
### Prepare your artwork

First, delete the cat! You should then import the two sprites, for the boat and gate. Since they're not in the Scratch 1.4 library, you can download them ([magpi.cc/scratch\\_art](http://magpi.cc/scratch_art)). Just click the star/folder icon above the Sprite List (bottom right), then navigate to the folder where you've stored the downloaded graphics for this project. Import the Boat and Gate sprites. If you aren't designing your own course, you can also download and import our Course backdrop: click Stage in the Sprite List, select the Backgrounds tab (top middle), then click Import and navigate to the folder.

## >STEP-02

### Design a course

You could just edit our course. Alternatively, to create a brand new one, click on the Stage in the Sprite List, then the Backgrounds tab, and Paint. Use the paint bucket tool to fill the canvas with a blue colour for the water. Then use a brown colour – which should be the same as in the Gate sprite – to draw the walls of the course. Use a yellow colour to draw some sand for the finish. Finally, add some white arrows which will act as speed boosters. Once this is done, let's make our Gate sprite rotate by adding the simple code in **Listing 1** to its Scripts area.



**Left:** We used touching color Sensing blocks to detect when the boat has hit a hazard, booster, or the finish



.03



## >STEP-03

### Controlling the boat

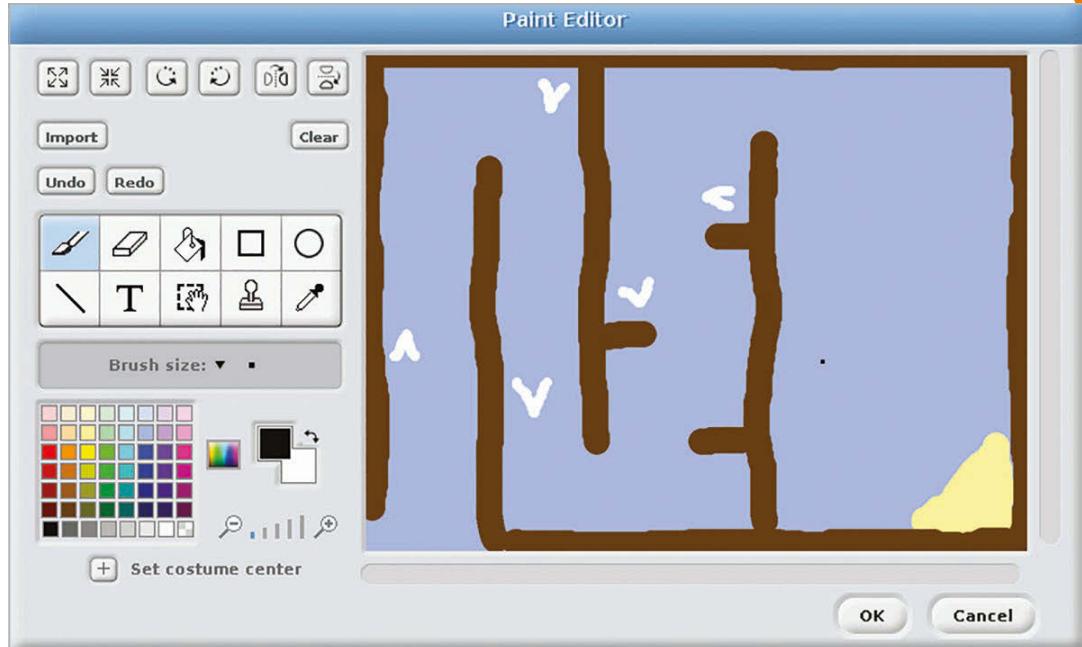
In this game we'll be controlling the boat with a mouse – using the code in **Listing 2** in the Scripts tab of the Boat sprite. To do this, we simply point it towards ‘mouse pointer’ and move it 1 step at a time, within a **forever** loop. To stop it from moving when near the pointer, we put the control code in an if block that only tells it to move if the distance to the pointer is greater than 5. Try out the code and guide the boat: at the moment, it sails straight through barriers.

## >STEP-04

### Make it crash!

What we need is some collision detection to check whether the boat has hit a hazard. Within your **forever** block, add the code from **Listing 3** under your boat control code. Here, we use the **touching colour** Sensing block to see if the boat has hit anything brown: click the colour square to get a dropper tool, then click on a brown part of the course. When it crashes, we switch the boat’s costume, say ‘Noooooooo!’, then place it back at the start point (in its normal costume).

Let's add two more **if touching color** blocks, shown in **Listing 4**, to our forever loop. The first checks whether the boat has reached the yellow beach, which acts as the finish line, and stops the program. The second detects the white of our booster arrows and moves the boat three steps.



**Left:** You can edit the course in the Paint Editor or create a brand new one

## >STEP-05

### Boosters and time

To make our game a bit more exciting, we need a timer. Click the Stage and add the **Listing 5** code to its Scripts area. This sets the time to zero at the start of the game, then gradually increases the **time** variable in line with real time – you'll need to create the latter in Variables and make sure it's ticked so that it's shown on the stage.

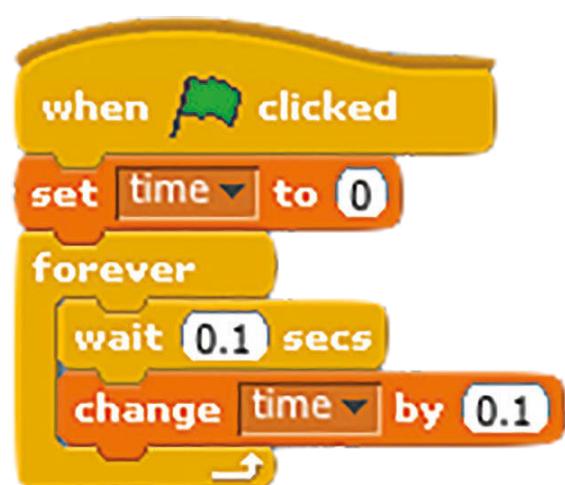
## >STEP-06

### Taking it further

You could easily add a sound effect for when the boat crashes, using a Sound block. You could even add background music, composing it using Sound blocks with various drums, instruments, and notes. The best time(s) could also be stored in a variable or list.



.04



.05