

# Scratch 1.4 Exercises

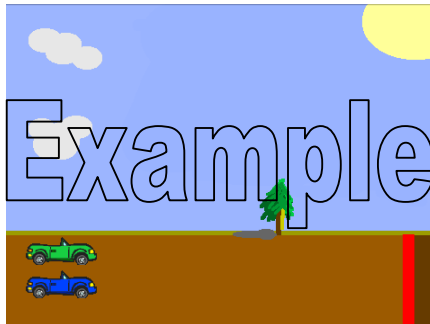
BY MR. REID

## Our Third Program: Two Race Cars

In our previous examples we learned about two important concepts sequence and repetition. We also learned about running two sprites at the same time using the “when flag clicked” control block. In the next programming example we will build a short program that contains two major parts. Part one I call the “**Visual Stage**”. In this stage we place the sprites, paint the background and setup our visual elements of the program. Part two I call the “**Programming Stage**”. In this stage we drag the instruction blocks into the script area to build our program. Each sprite is programmed separately.

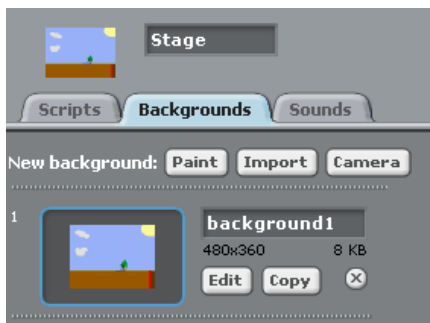
### The Visual Stage I

In this first stage we are going to draw a simple dirt road and sky picture like the one you see here.



Step 1: Run Scratch and click on the stage found in the **sprite list** section.

Step 2: Click the **Edit** button and use the drawing tools to paint the background. Include any details you think are important. You will notice in the picture to the left that the background in the example is already painted.



It is really important to play with the different drawing tools. Take your time and be creative. You may notice my tree has a shadow. *Change your brush size too!*

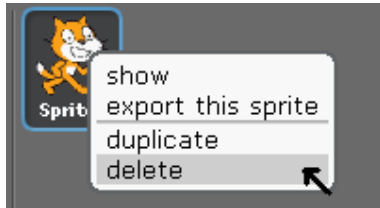
Your background must have a red finish line!



**Drawing Hint:** I use the ellipse tool and light grey to create the clouds. Just overlap the shapes to make fluffy clouds.

## The Visual Stage II

Now that you have painted the stage area, we are ready to add a car. We will paint the car too!



Step 1: Delete the cat sprite that comes with every new project. We will not be using the cat in this program. To delete this sprite right click and choose the delete option from the pop-up menu.



Step 2: Choose the “new sprite from file” option.



Step 3: Choose the “transportation” folder and then choose the green color car. The car is titled “car1”.

There are other car choices in this folder. If you like pick a different car for your project. It must be a car for this program.

Step 4: Select the car1’s costumes tab from the **sprite information** section.

Step 5: Select the **Edit** button and paint the car blue. Use the fill tool. Make light green areas light blue, and dark green areas dark blue.

Step 6: Paint the tires dark grey using the fill tool.



Above you see your finished car. Notice that the handle and under carriage of the car are dark blue. If you make a mistake you can always use the undo button to go back and try again. Take your time and have fun!

Step 7: Shrink your sprite, I clicked 25 times



small enough. sprite using the “shrink or more to make my sprite

Step 8: Drag your car sprite to its starting position on the stage.

## The Programming Stage I

Now we are finally ready to start programming. Wait! Where is the second car? Don't worry this part is tricky, so we are going to take it one step at a time. We will program the blue car first and test our code using only one car. Building a small working part of our program first is always a good programming practice when the project is complicated.



Step 1: Drag a single move block called "set x to \_\_\_" onto the script area of the blue car.



Step 2: Hold your mouse pointer over the center of the car sprite on the stage. Enter the x coordinate value into the white space of the "set x to \_\_\_" block.

Step 3: Click this block once to execute (run) it. The car sprite may "move" a little. This is your car's starting position in the "race". If you don't like this position, change it to suit your tastes.



Step 4: Drag a move block onto the script area.



Step 5: Combine the move block with a repeat block. Place the move block inside the repeat block. This is called **repetition**. It means that a set of instructions should be repeated (done more than once).

Experiment by clicking the "repeat until" block of code. What happens? To stop the code, press the red stop button at the top of the stage.

Try clicking on the "set x to" block of code. What happens? There are other code blocks that repeat instruction blocks. Try different blocks and see what happens. Write down your observations.



Step 6: Select "touching color" from the sensing pallet. Drag it into the "repeat until", evaluation area.

Step 7: click on the color square, and using the mouse select the brown area beyond the red "finish line". This color must be different than the brown color the car starts on. What happens when you run the block of code now?

## The Programming Stage II

If you have been successful you now have two blocks of instructions. One block is a single “set x to \_\_\_” block that when clicked resets the car to its starting position. The other block of code is a repetition block that will drive the car to the other side of the stage, until the car is touching or just passed the read finish line.

At the moment your car simply moves forward 10 steps every time the loop executes. In other words, the car is moving as if you kept clicking your mouse on the move block, “until” the car touches the finish line.



Step 1: Add a “Say” block from the “looks” pallet and change the text to read “The Blue Car Wins!”.

Step 2: Add a second sprite to your program. I will assume you choose the green car so that you will have a green and blue car. However you may add any car you like to this project or you may change the color of the second car.

Step 3: Copy the two blocks of code from the blue car, to your new car. Remember to change the “Say” block to refer to your new car.



Step 4: Add a “when flag clicked” block to the top of each repeat block, one for each car.

You will notice that running this program is not very exciting. What happens?

### Random Numbers:

Many video games make use of random numbers to make programs respond differently each time they run. The computer chooses a number from a set of values and uses this number in our programming block below.



Step 5: Add a pick random number block from the “operators” pallet. Place this inside the move block where the value 10 is located, for both cars.

Step 6: Place the “set x to \_\_\_” block just after the “when flag clicked” block, for both cars. This should leave you with a single block of instructions for each car.

Think of ways you could improve this program!