



# Code a Car Racing Game

MYO Spring 2022





# Volunteer Introductions

- Name
- Preferred pronouns
- Year
- Major
- Why I chose STEM?
- Post-graduation plans

# New To Coding?

- Have you ever seen a code?
- Programmed something yourself?
- Never seen a code before?
- We all start somewhere!

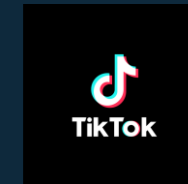
# Computer Science Is Everywhere!

Related Majors:

- Computer Science
- Computer Science and Engineering
- Computer Engineering



Most majors and professions use computer science or coding in some way!



```
Final Project.ipynb
File Edit View Insert Runtime Tools Help Last edited on Dec 12, 2020
+ Code + Text
from __future__ import absolute_import, division, print_function, unicode_literals
import tensorflow as tf
import numpy as np
import os
import matplotlib.pyplot as plt
import pandas as pd
import csv

[ ] from google.colab import drive
drive.mount('/content/drive/')

Drive already mounted at /content/drive/; to attempt to forcibly remount, call drive.mount("

[ ] Is '/content/drive/My Drive/Senior Year/BME3520'

'Final Project.ipynb' HAM10000_metadata.csv images/ 'Lecture11 (1).ipynb'

[ ] base_dir = '/content/drive/My Drive/Senior Year/BME3520/images'
```

Can you name any examples?



# Today we are making a car racing game

We will be using a website called Scratch.

Scratch is a coding language with a visual interface

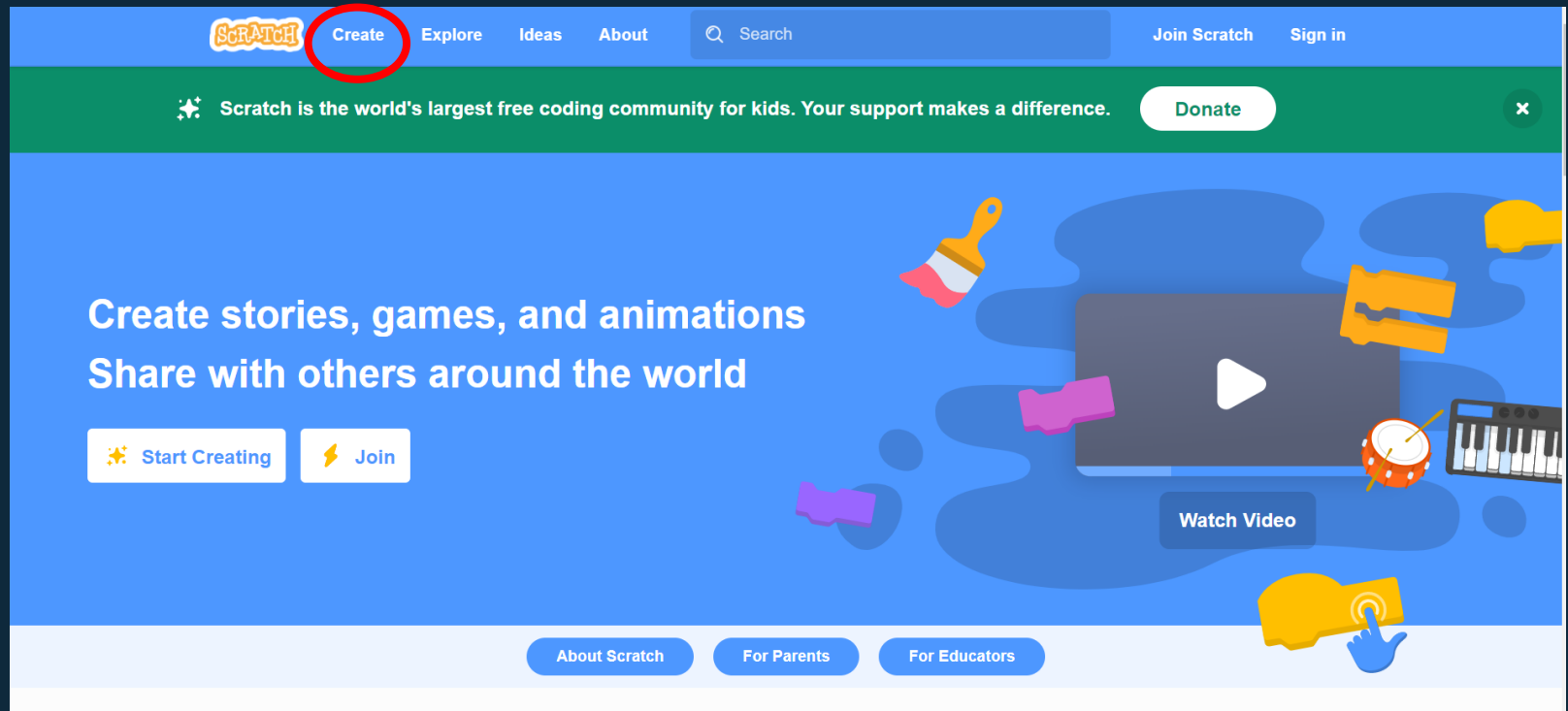
\*Interface: where interactions between humans and machine occur

Let's see what our game will look like!

# Let's Get Started!

Go to [scratch.mit.edu](https://scratch.mit.edu)

Click “Create” at the top



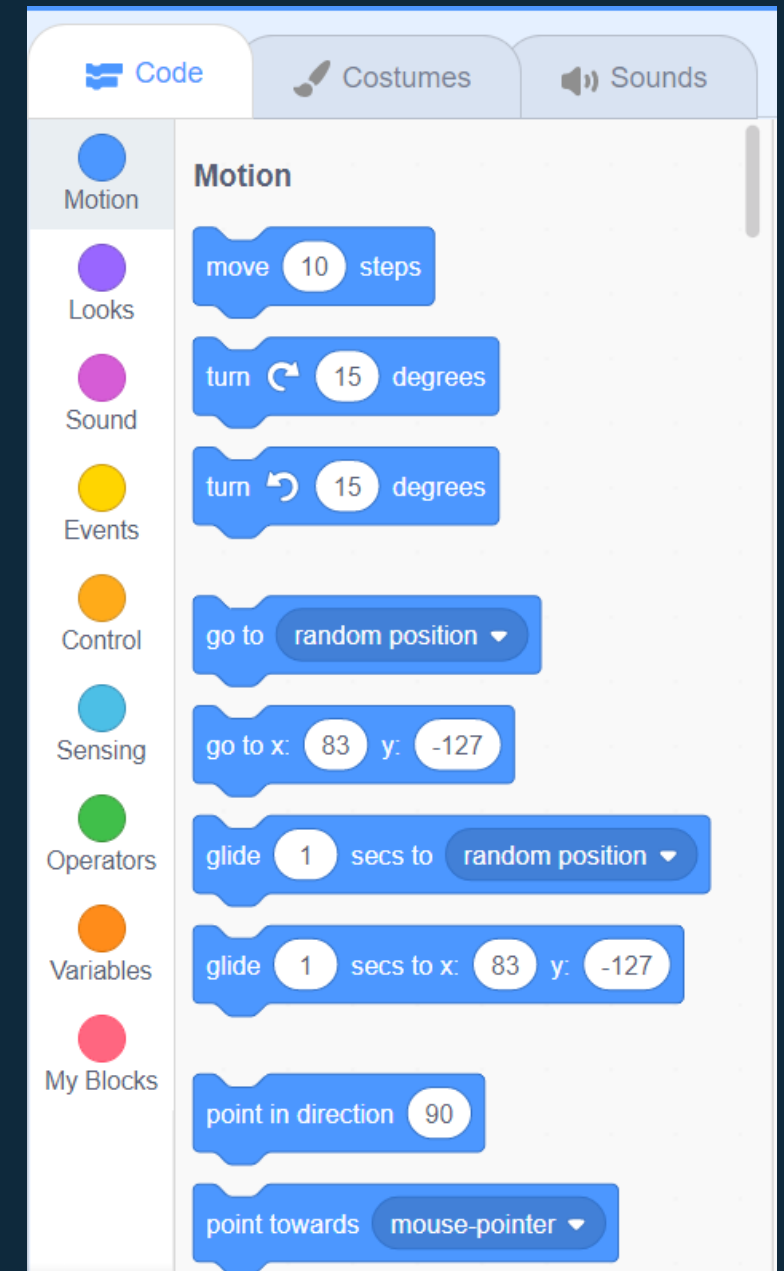
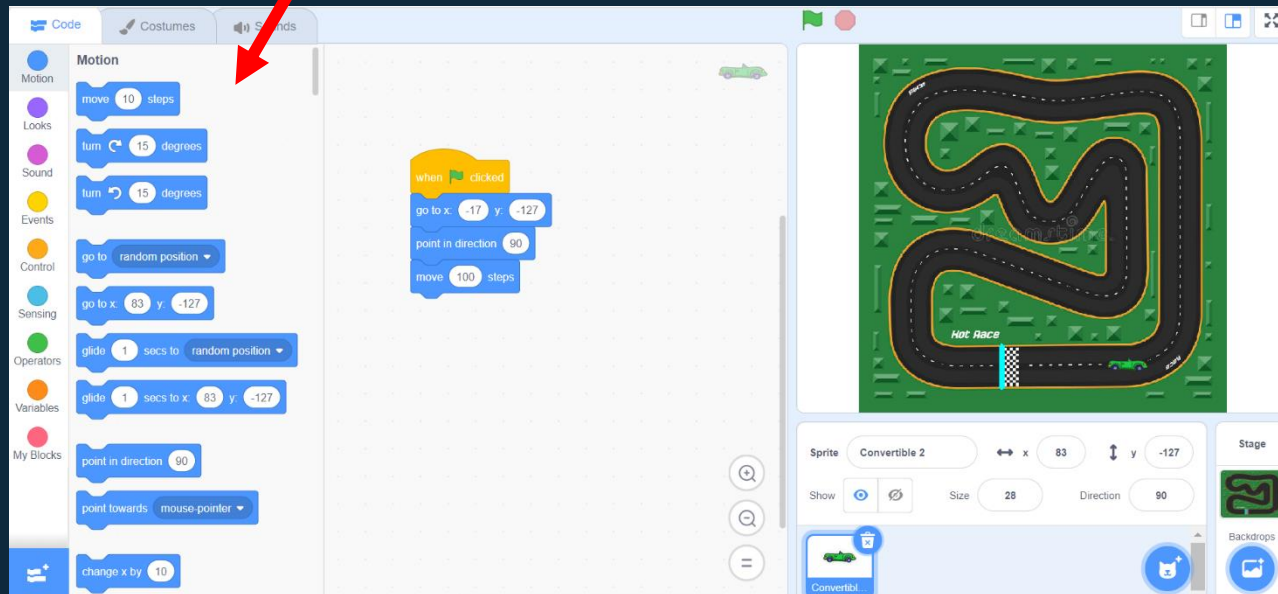
# Home Page

The image displays the Scratch IDE interface. On the left is a sidebar with category icons: Motion, Looks, Sound, Events, Control, Sensing, Operators, Variables, and My Blocks. The main workspace is divided into three panes: Code, Costumes, and Sounds. The Code pane shows a script for a green car sprite:

```
when green flag clicked
  go to x: -17 y: -127
  point in direction 90
  move 100 steps
```

The Stage pane on the right shows a green background with a black racetrack labeled "Hot Race". A green car sprite is positioned at the start of the track. Below the stage, the Sprite area shows "Convertible 2" with a size of 28 and a direction of 90. The Backdrops area shows a single backdrop.

# Command Center: Motion

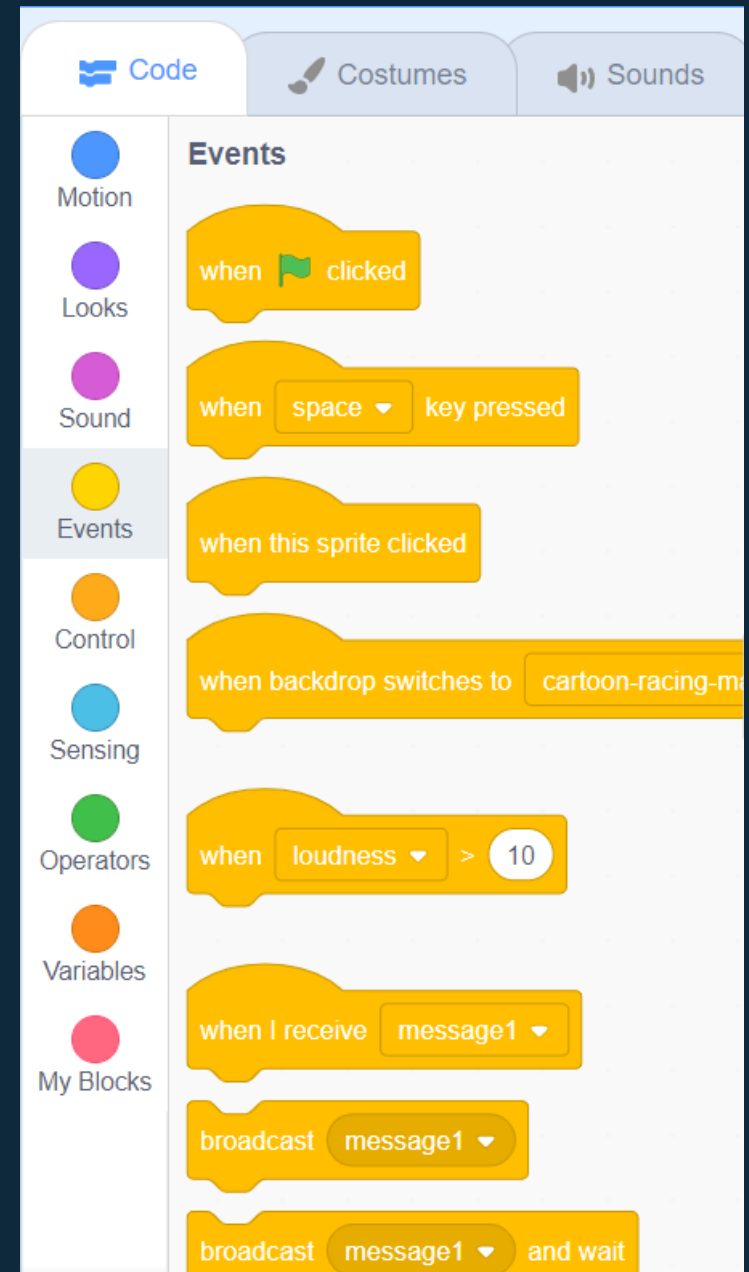






# Command Center: Events

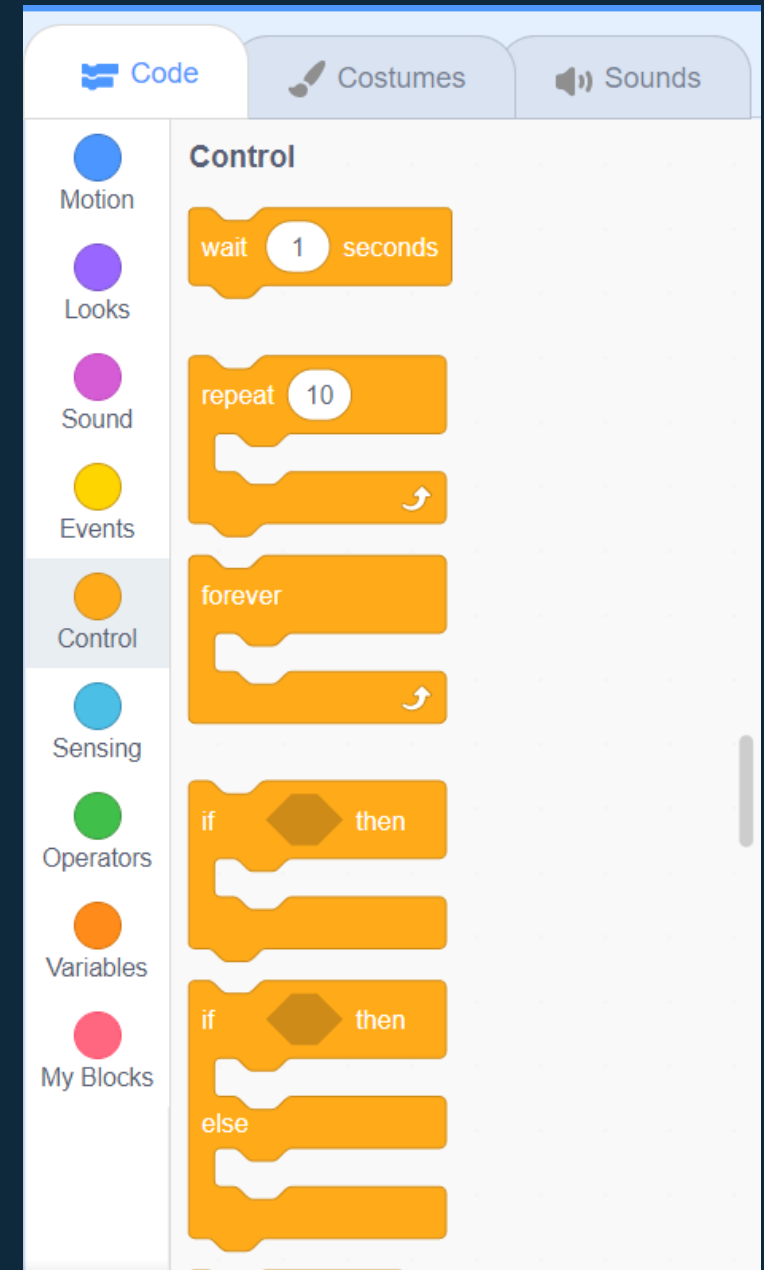
- Events go at the beginning of any code in Scratch
- You can have multiple events in one script
- The green flag at the top is used to start any code written





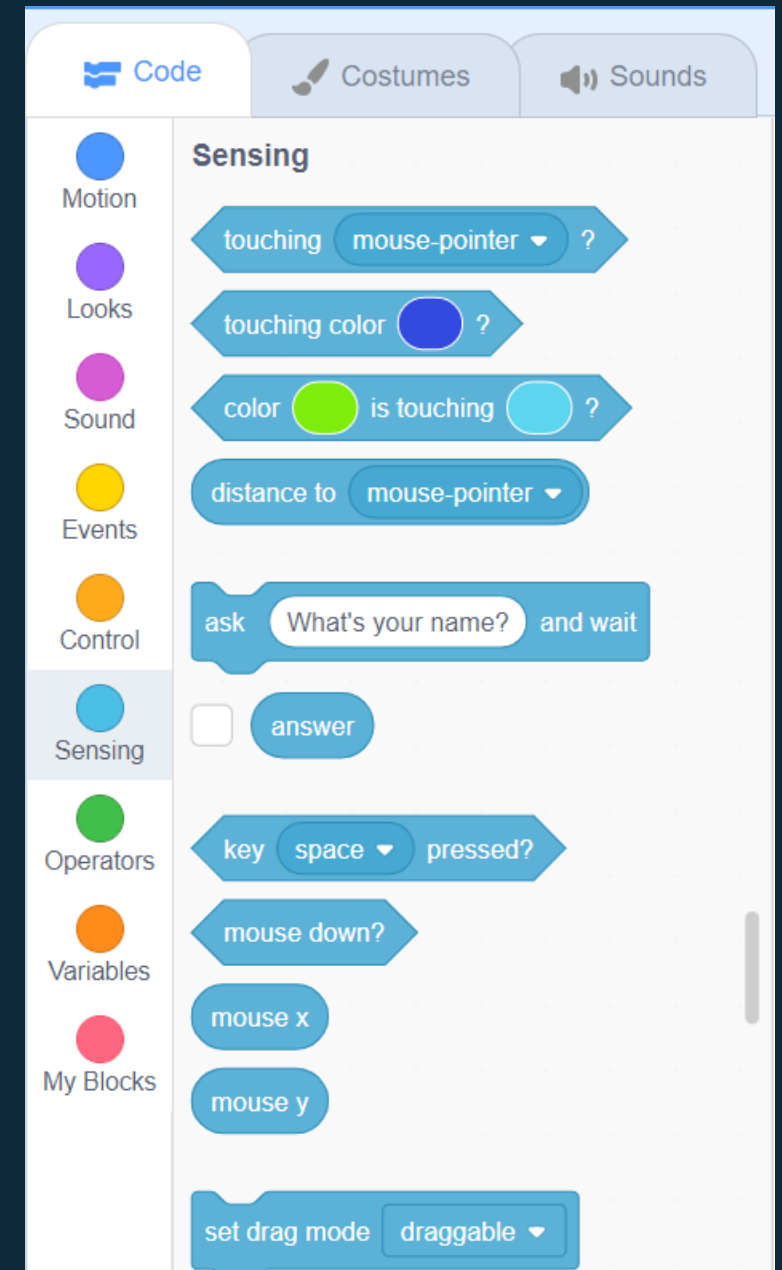
# Command Center: Control

- Controls specify when and how long a command is run
- You will learn about some of these control commands later

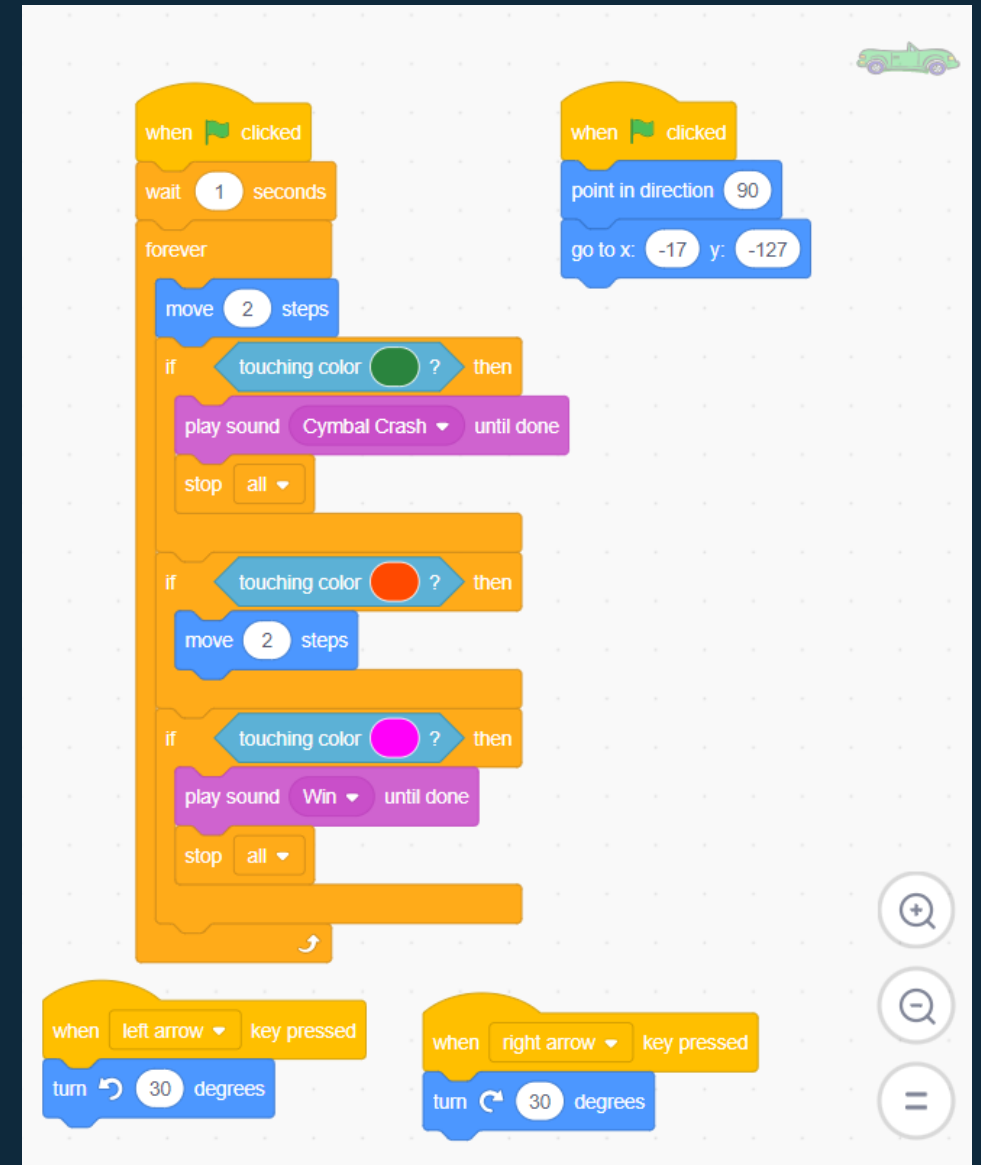
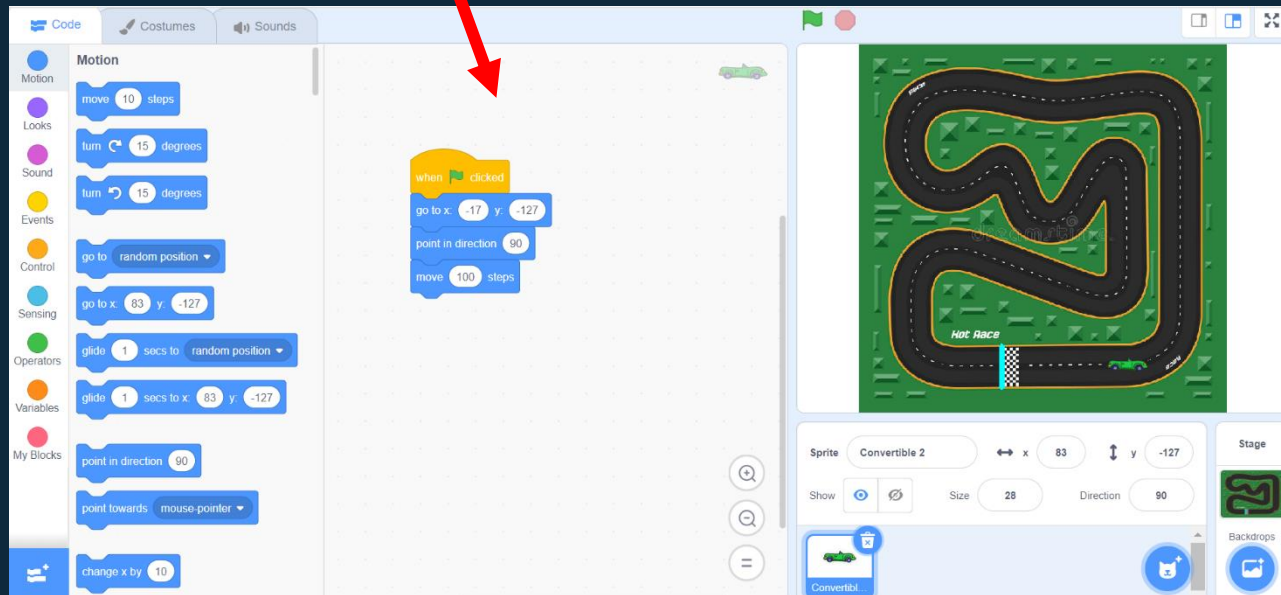


# Command Center: Sensing

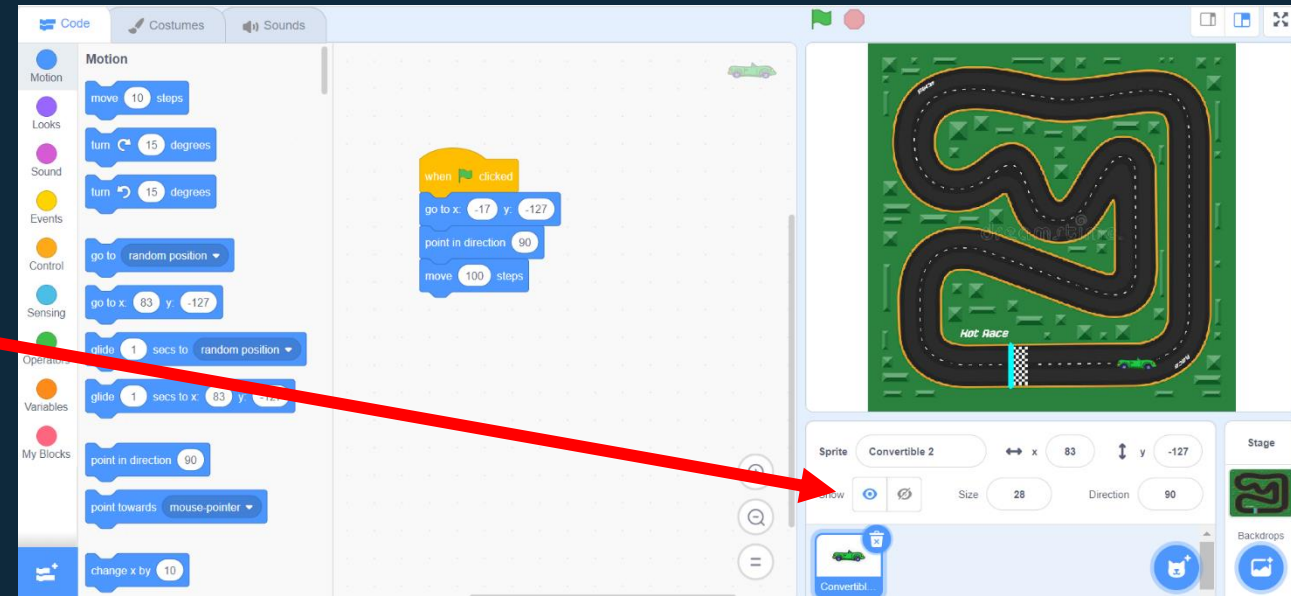
- Sensing commands pick up when something changes in your code
- Example: if object 1 moved closer to object 2, sensing commands would detect the movement and change how your code runs



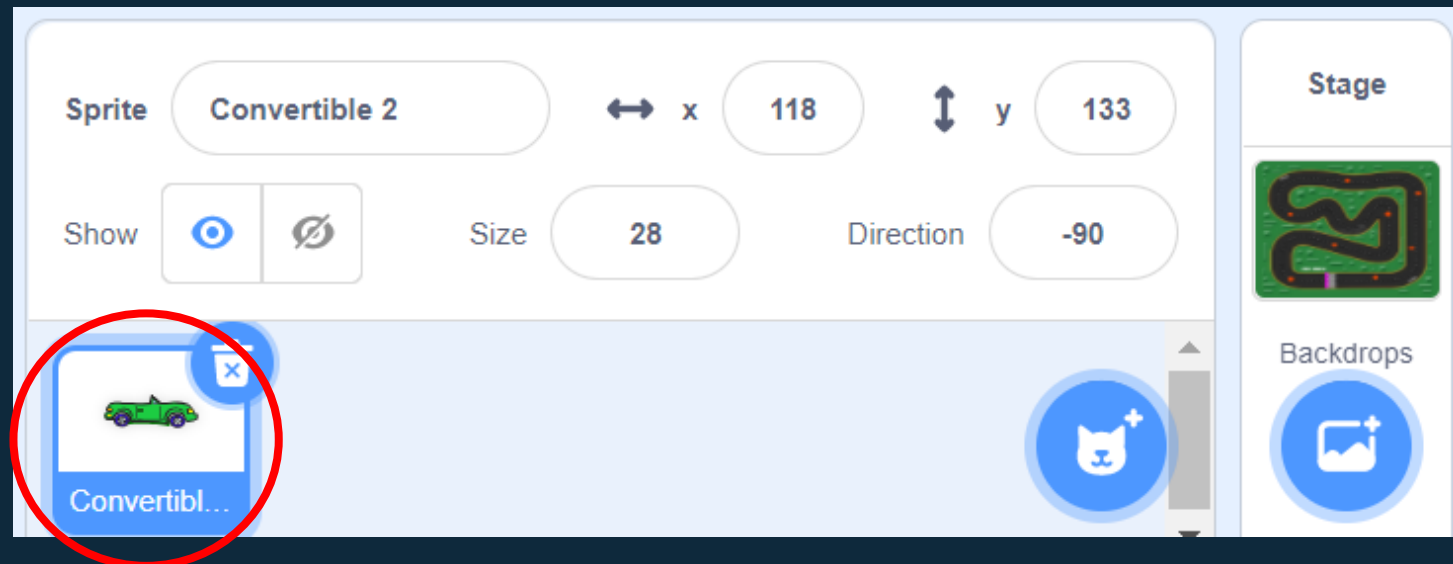
# Programming Area



# Costume Design Shop

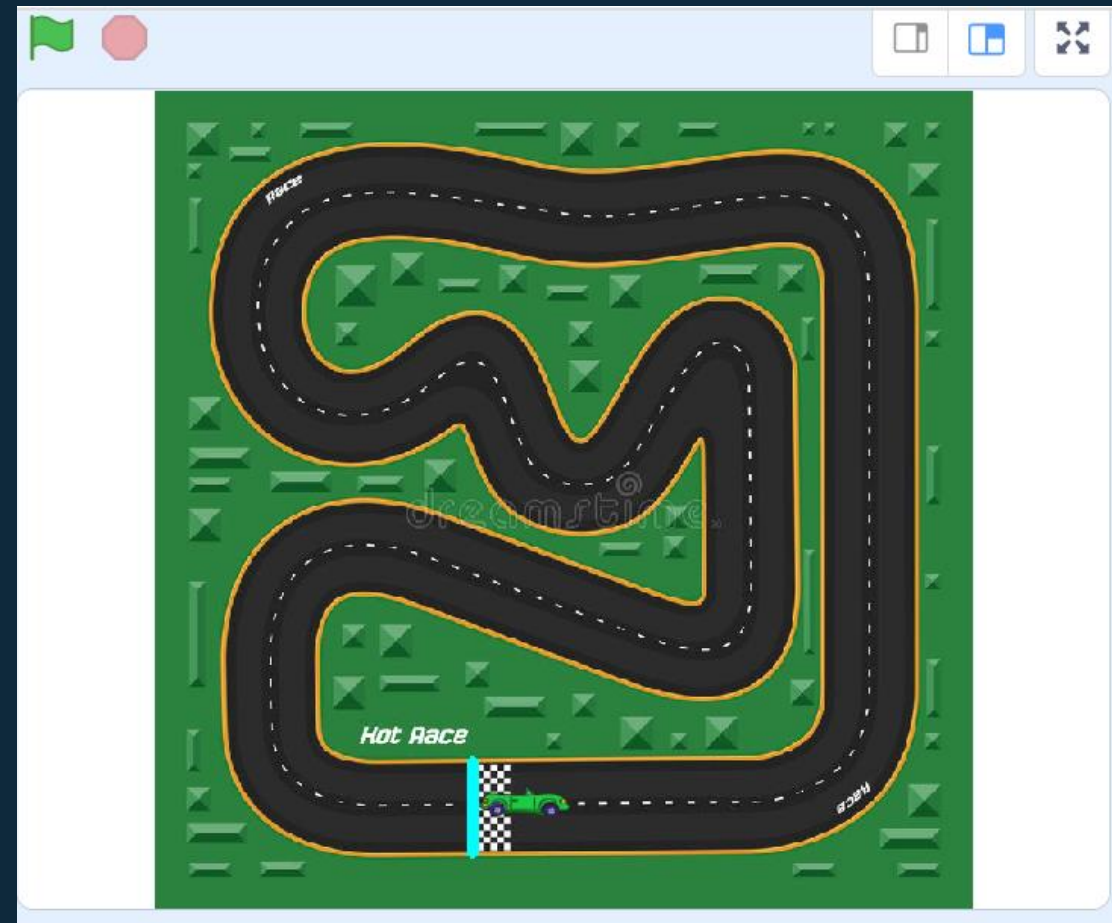
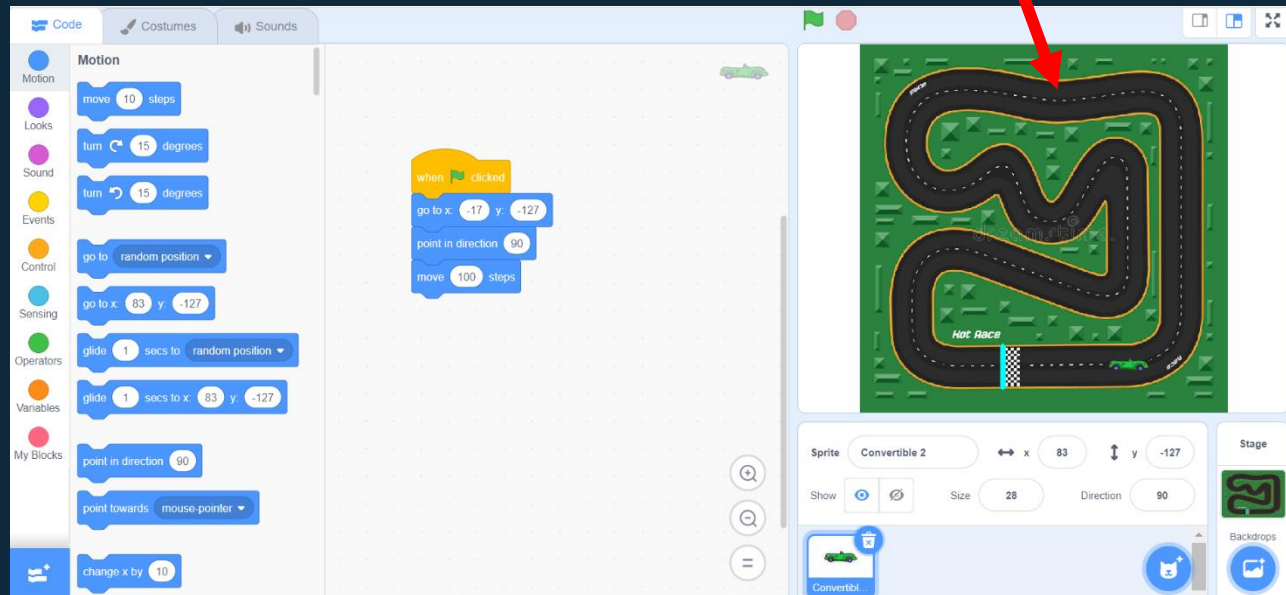


This is a Sprite

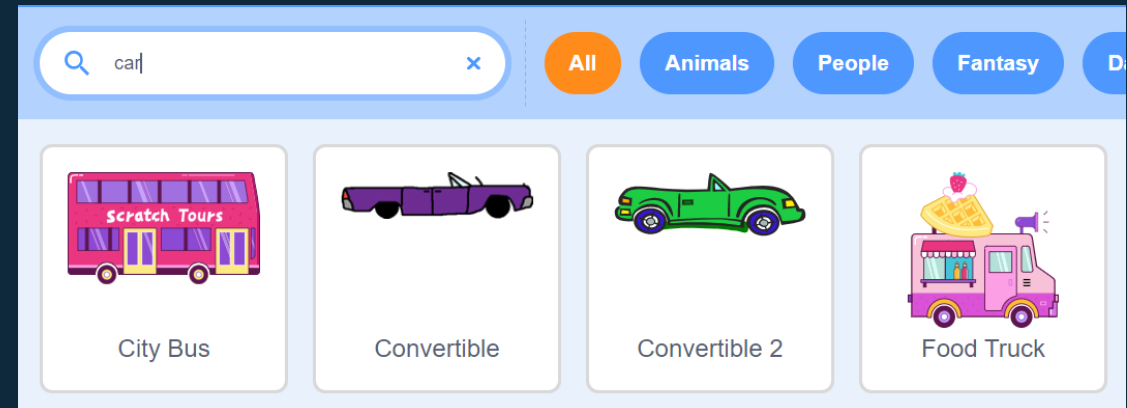
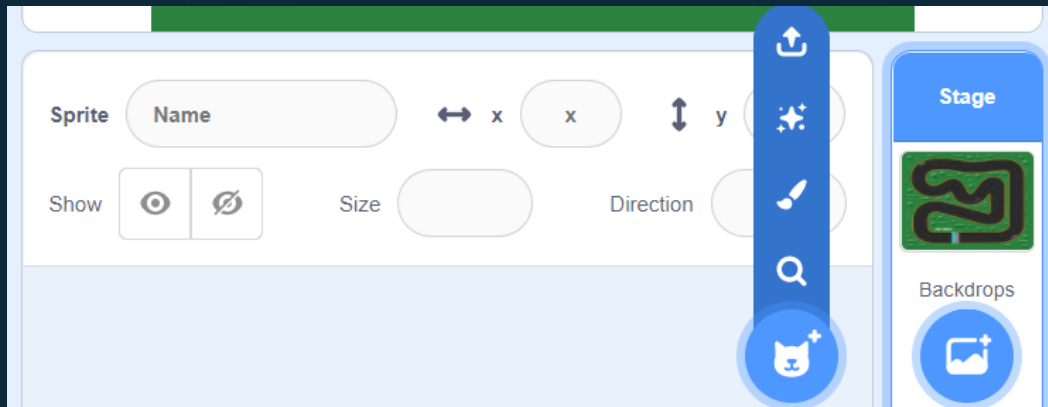




# Display Window



# Add Your Sprite



A decorative graphic on the left side of the slide. It features a large cyan hexagon in the center. Surrounding it are several smaller hexagons and icons: a lightbulb, a thumbs up, a smartphone, a magnifying glass, a gear, and a speech bubble. The background is a dark blue gradient.

# What issue do you see?

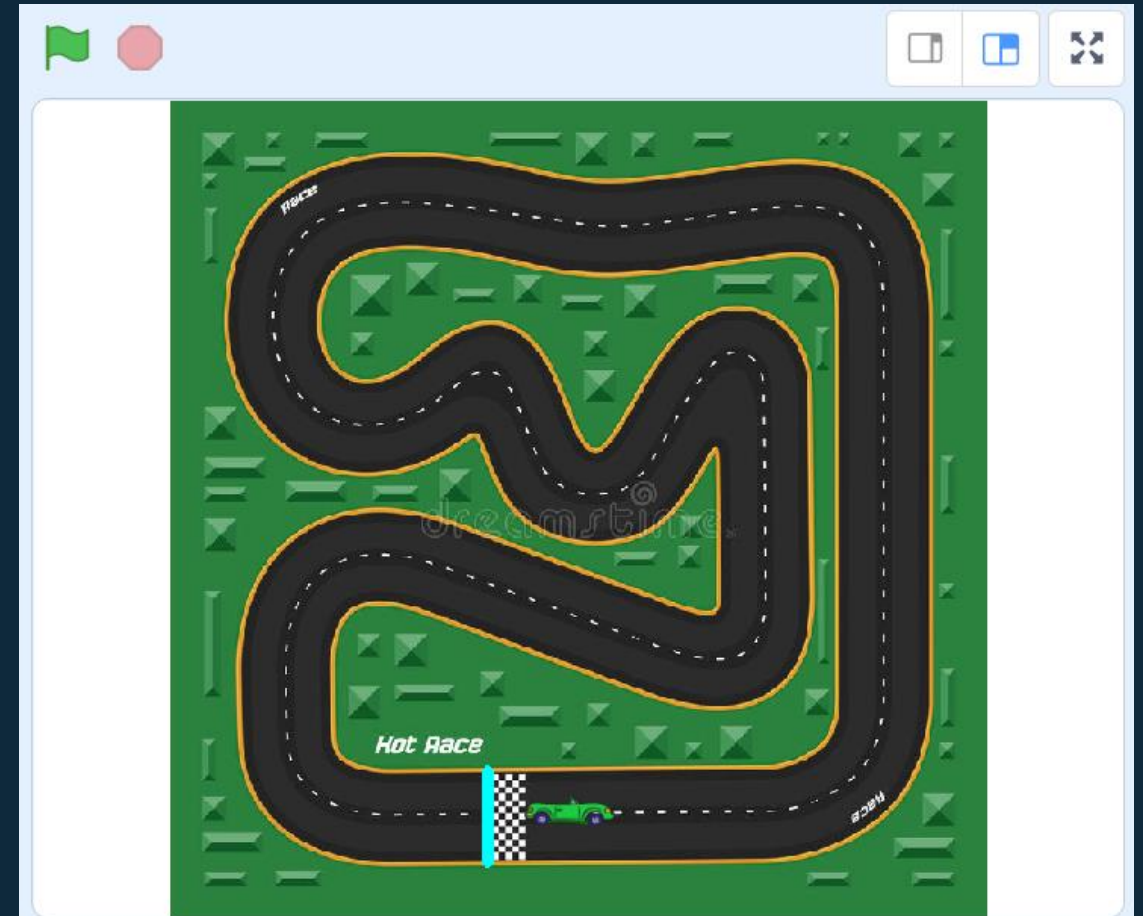
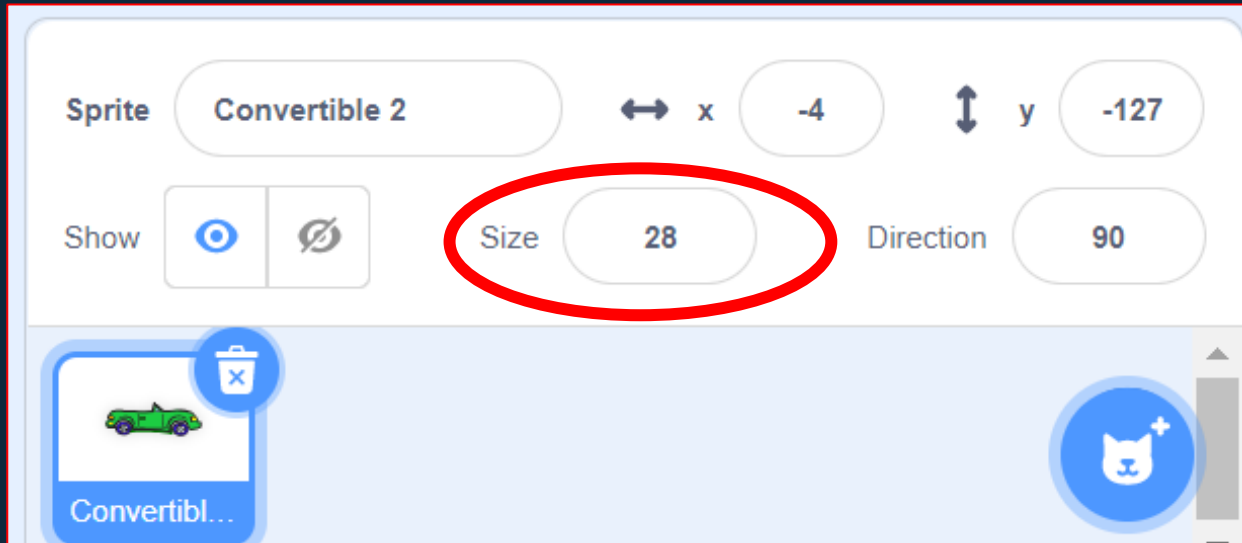
The car is too big!

## How do you think we can fix it?

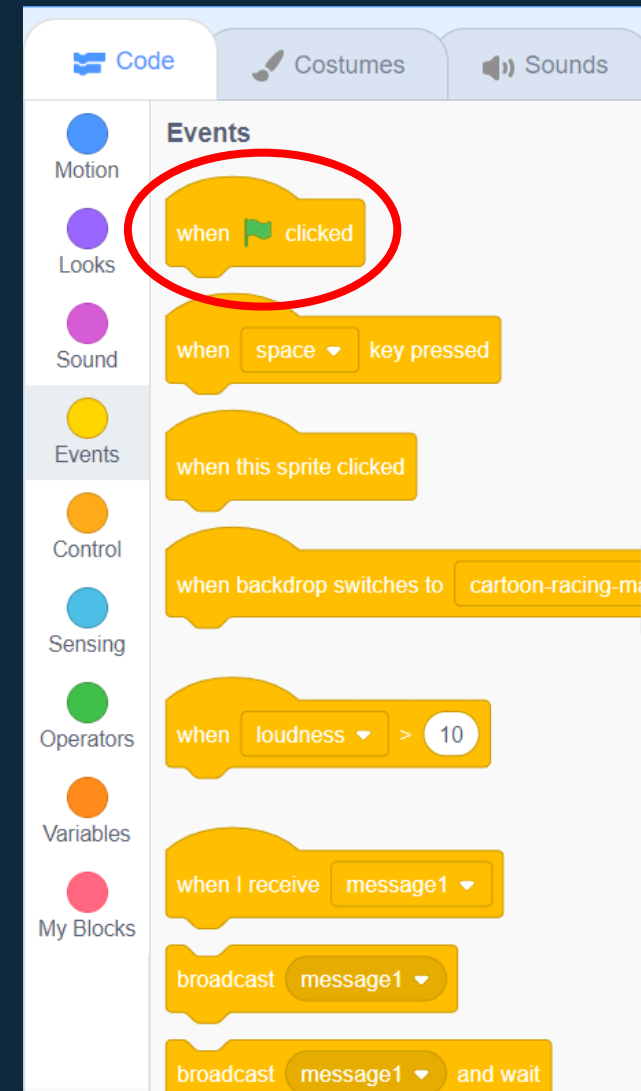
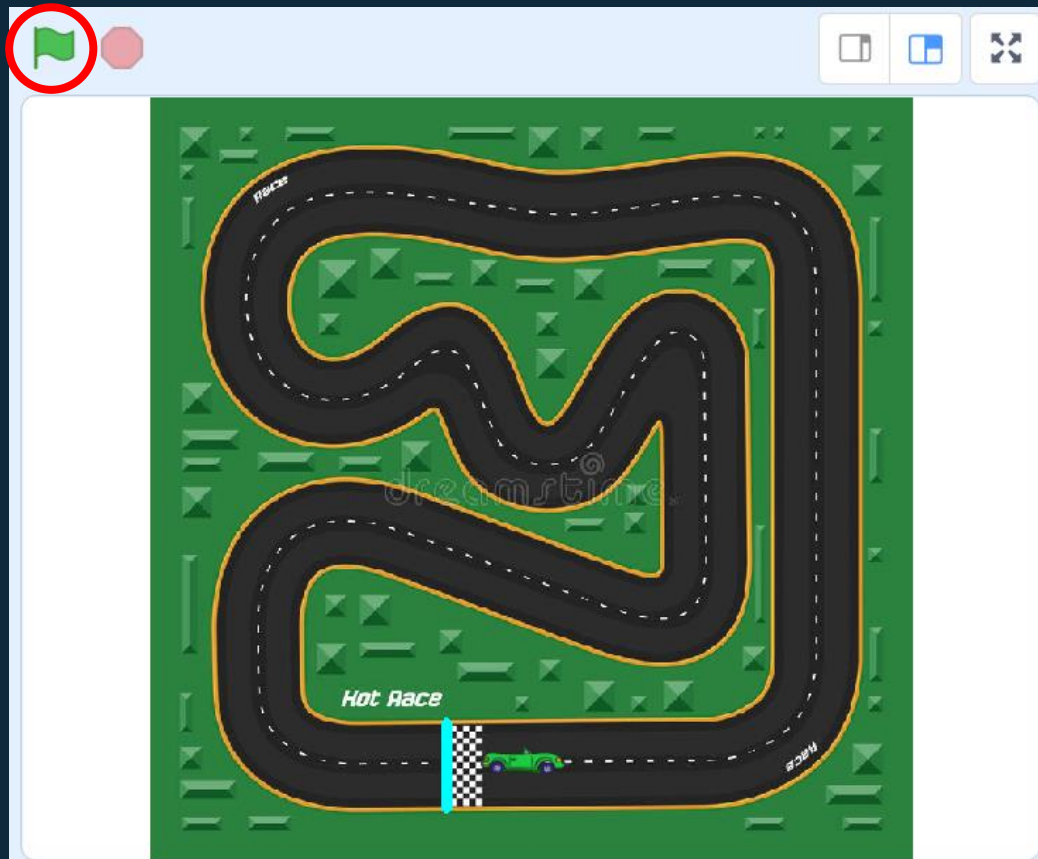
Change the size in the costume design shop!



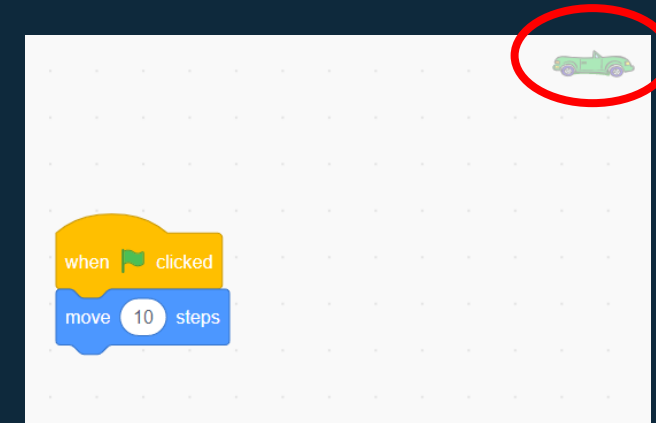
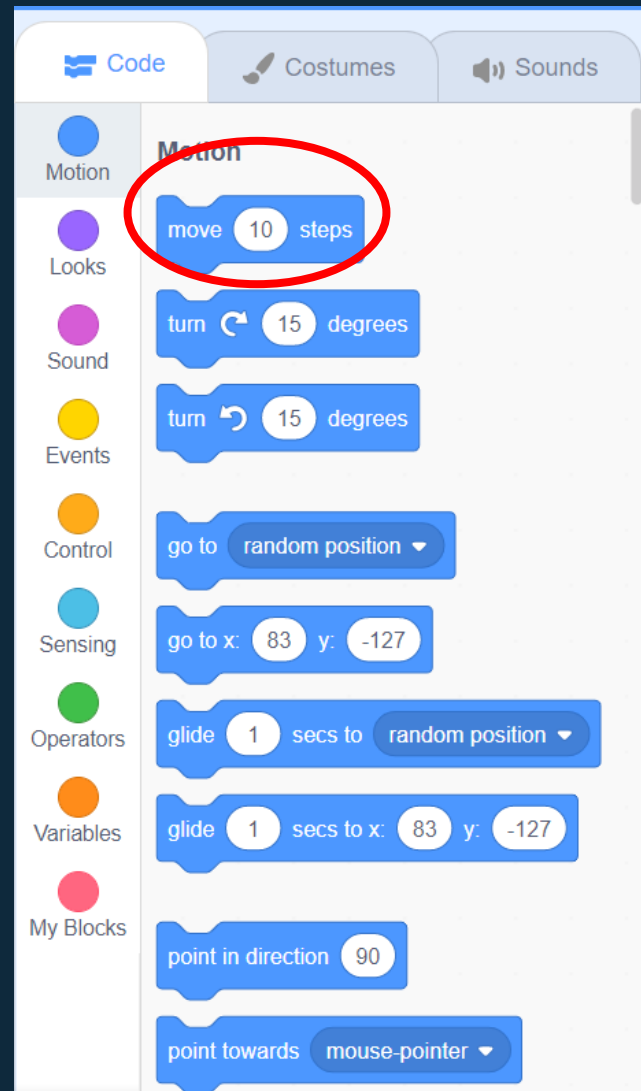
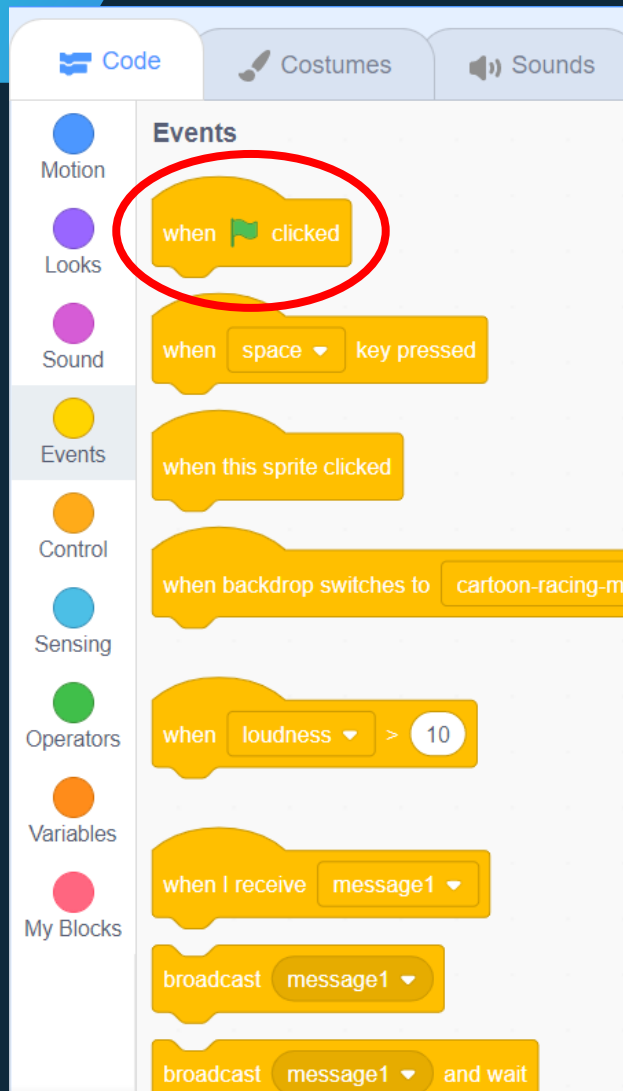
# Set Up Your Sprite



# Starting Your Code



# Programming Your Sprite





# What happens when you click the green flag?

The car moves 10 steps and stops

## How can we make it move continuously?

Use a loop!

# What Are Loops?

Sometimes you want a section of your code to run multiple times without having to press start each time

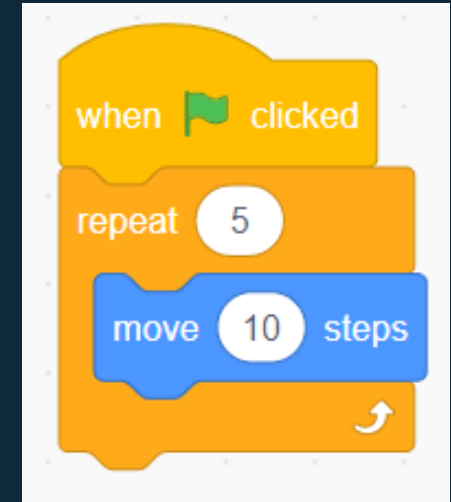
You can ***loop*** this section of the code to repeat on its own

Repeat X times:  
Move 10 steps

# Loops In Scratch

Repeat # Loops:

Will run the inside section of code X amount of times (5)

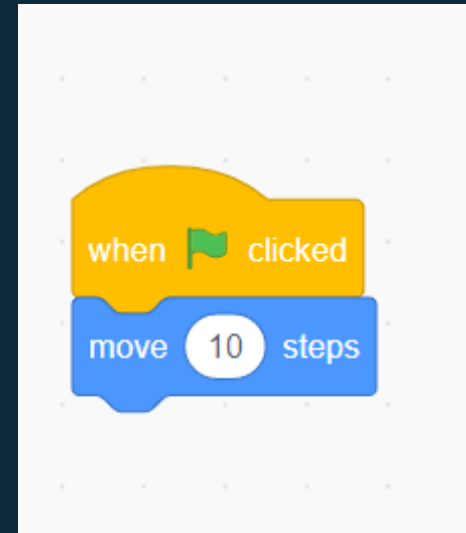
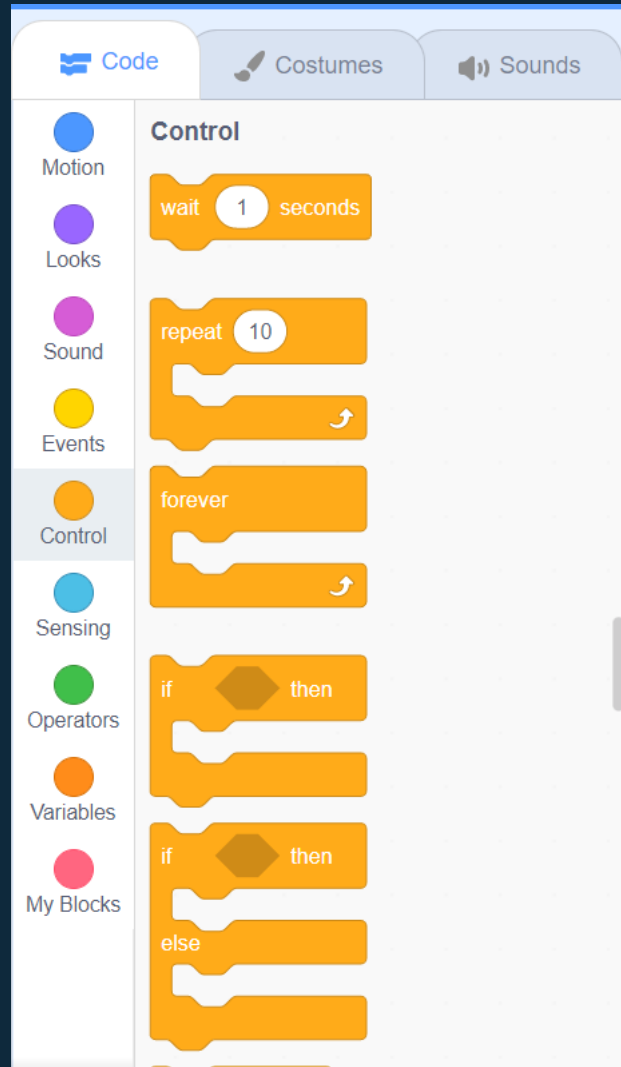


Forever Loops:

Will run the inside section of code forever **until the code is done**



# Add a Forever Loop





# What happens when you click the green flag?

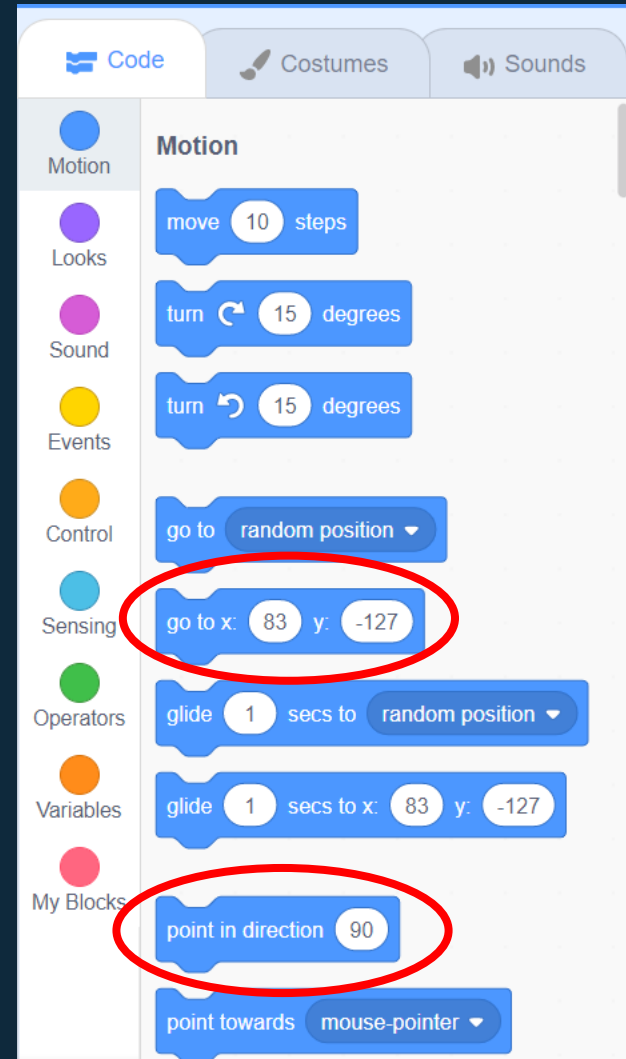
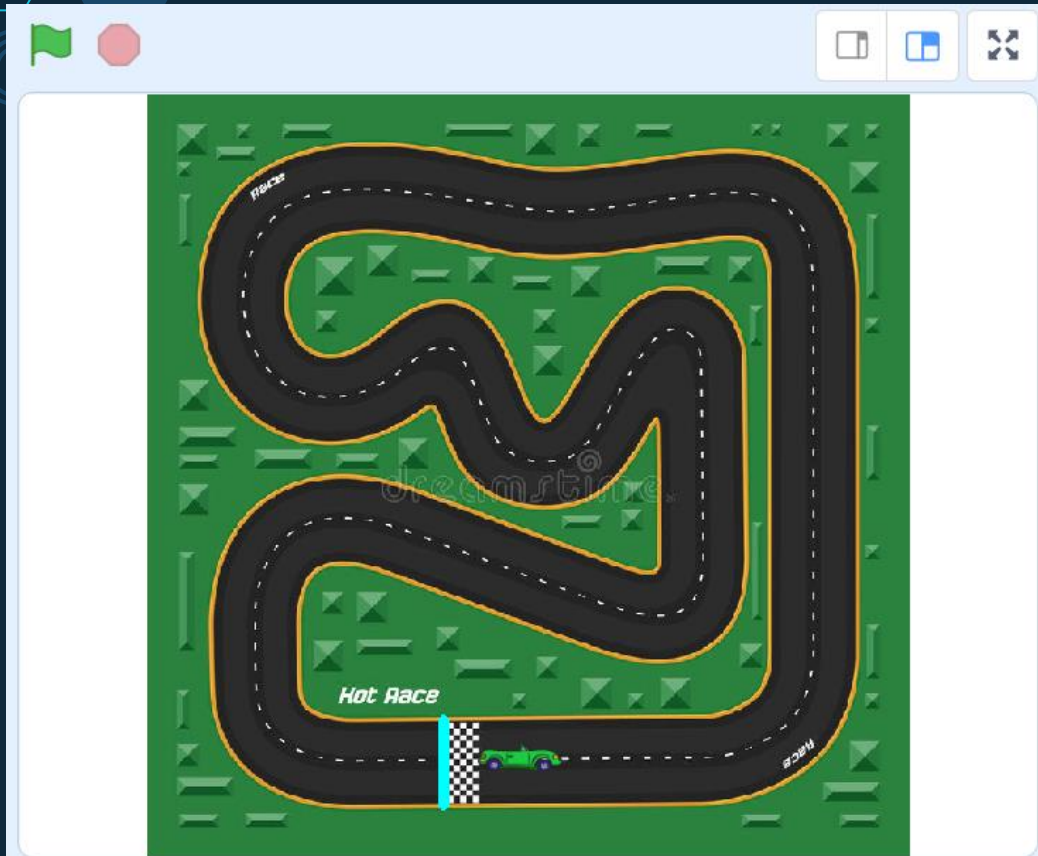
The car moves and doesn't stop!

# Try to run your code again. What happens?

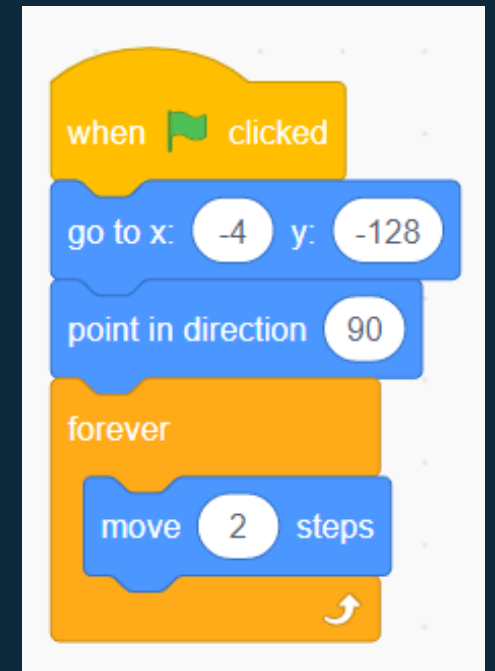
The car starts from the ending position of the previous run.



# Restart Your Sprites Position



Inside or outside  
the forever loop?



Outside!

A decorative graphic on the left side of the slide consists of a cluster of hexagons in various shades of blue and cyan. Some hexagons contain white icons: a lightbulb, a thumbs-up, a smartphone, a magnifying glass, and a gear. A network of dots and lines is also visible. A large, solid cyan hexagon is positioned in the center of this cluster.

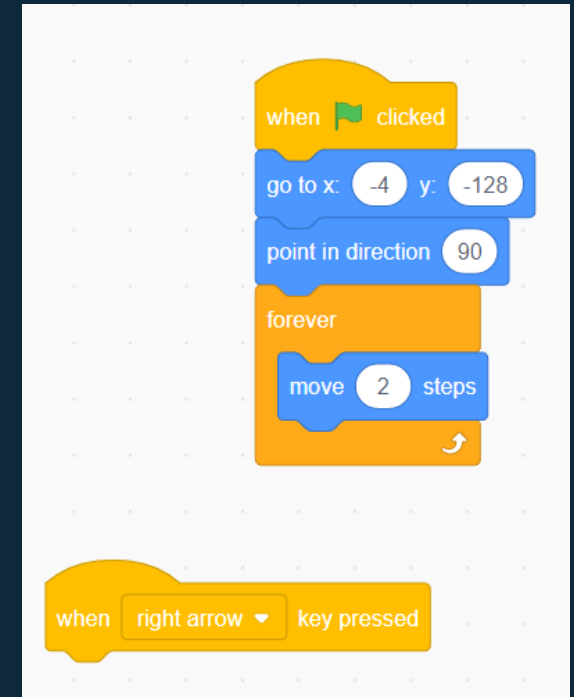
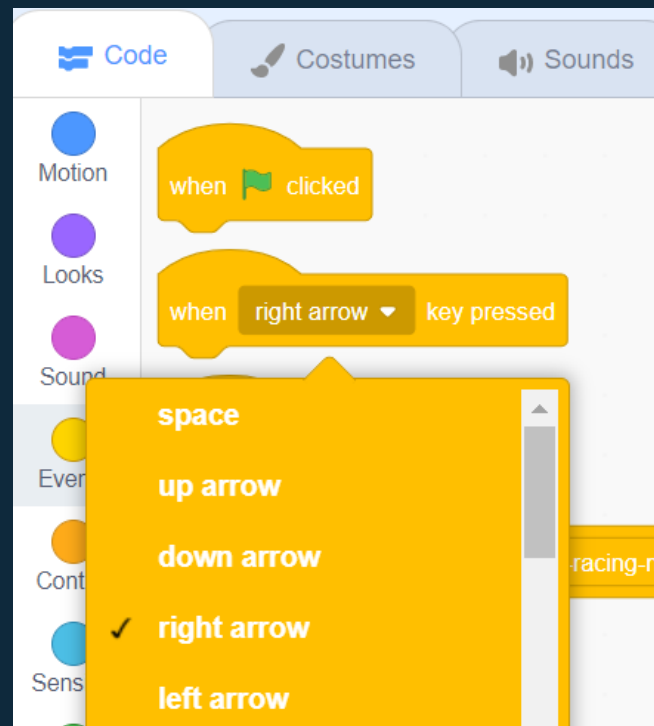
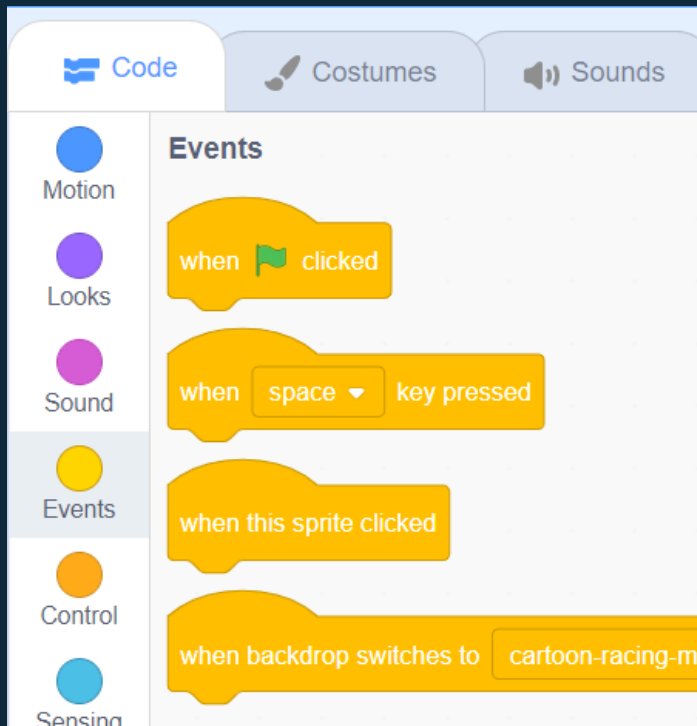
# Test your code!

Coding is always trial and error! Testing it after each step can help you find any issues.

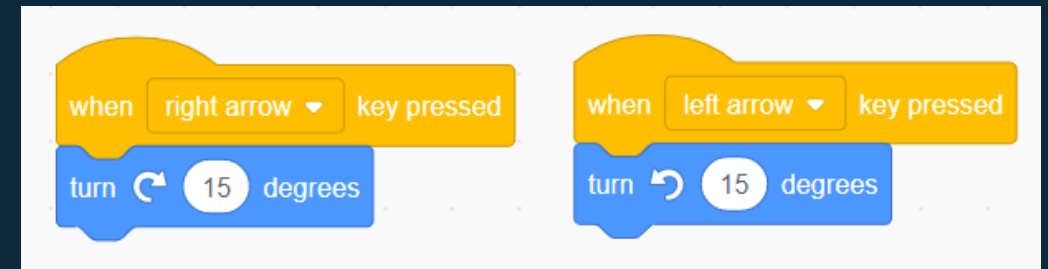
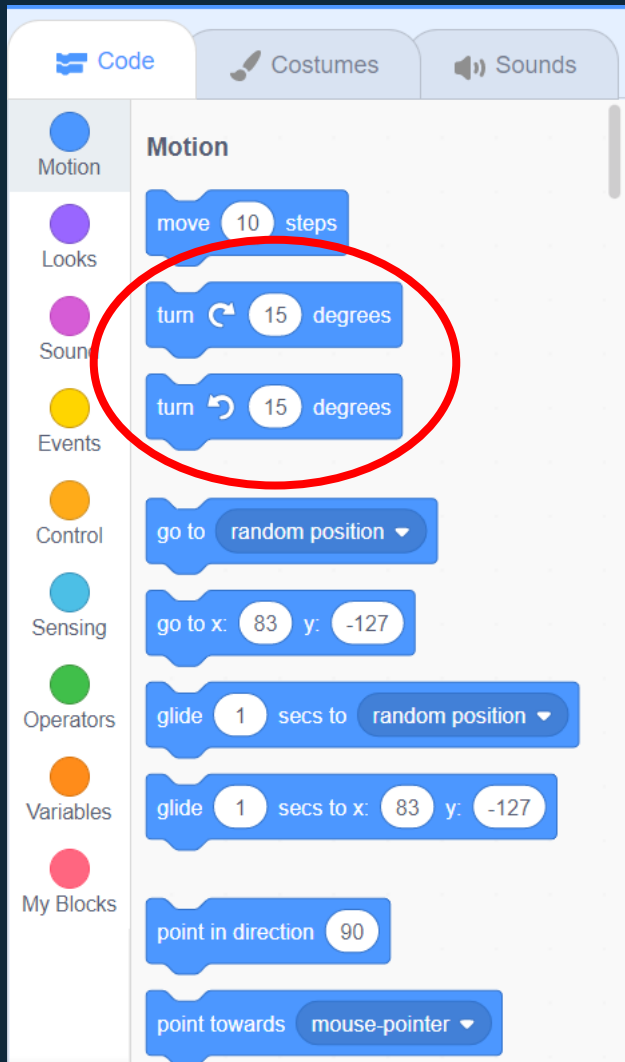
## Let's make the car turn!

# Set Up Buttons To Turn Your Sprite

Repeat this twice, once for turning LEFT and once for turning RIGHT.



# Choose Number of Degrees To Turn



Test out different numbers and find the one you like best!



# Test your code!

Does it work as expected?

## How can we make the game stop?

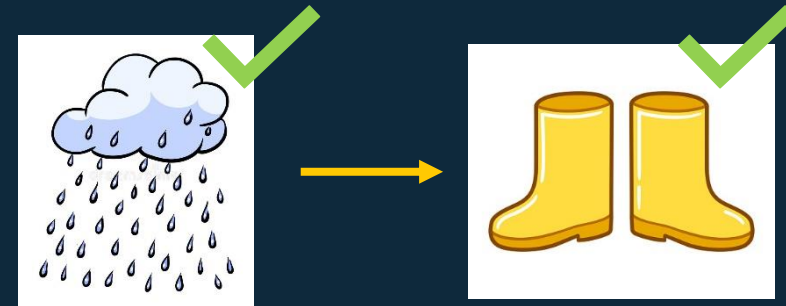
Either the car goes off the road or it crosses the finish line!

# Conditional Statements: If

When an instruction in your code depends on a certain ***condition*** being true, it can be written out using “if \_\_\_, then \_\_\_” statements.

Everyday example:

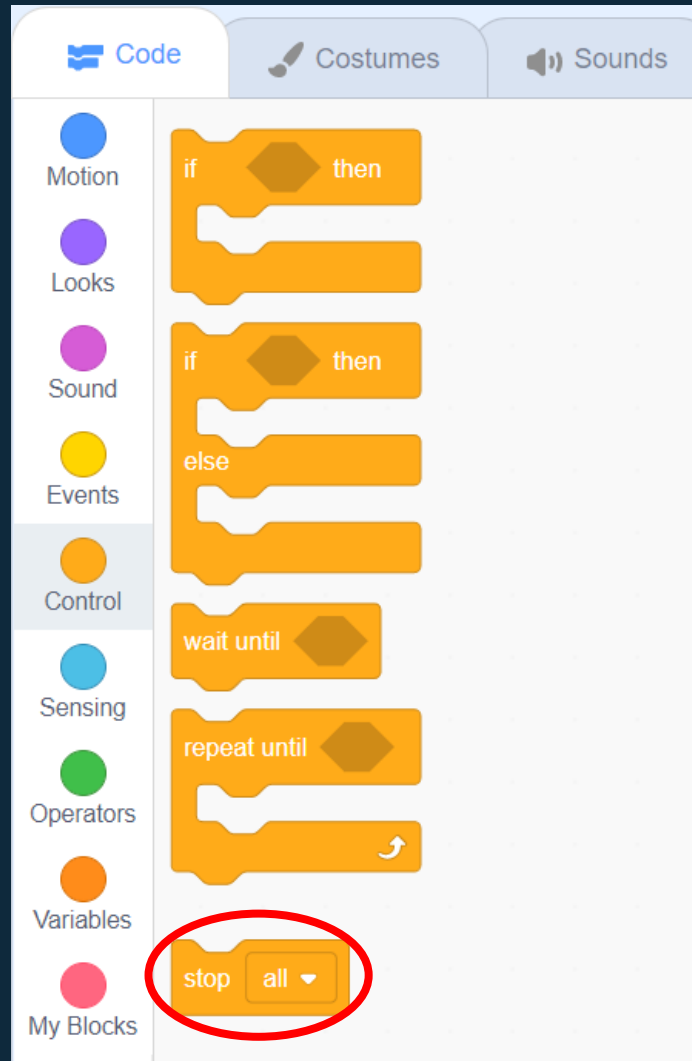
If it is raining outside,  
**Then** I will wear rain boots.



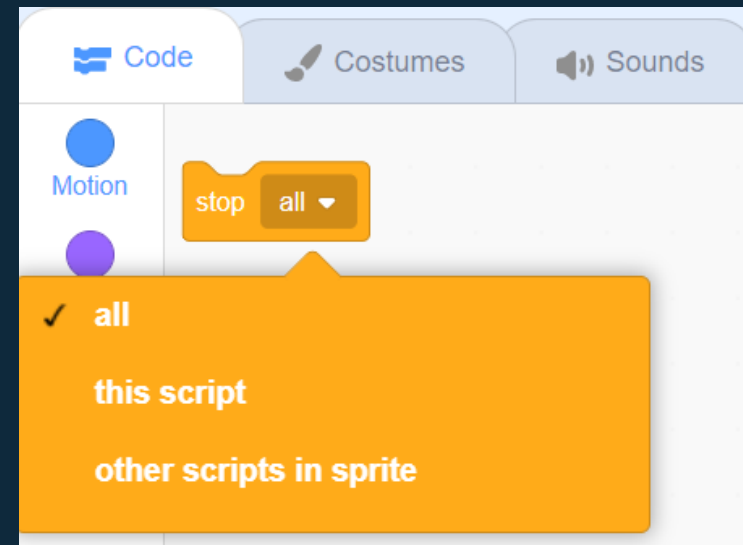
Coding example:

If my Sprite crosses the finish line,  
**Then** end my game.

# How To End Your Game



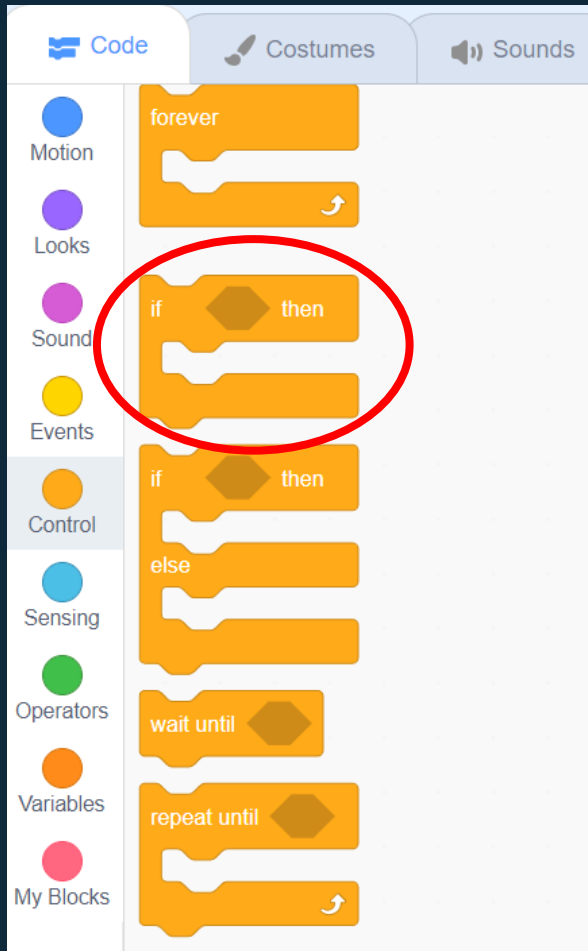
The “stop all” call will go inside an if statement.



For this game, leave it marked as “all”.

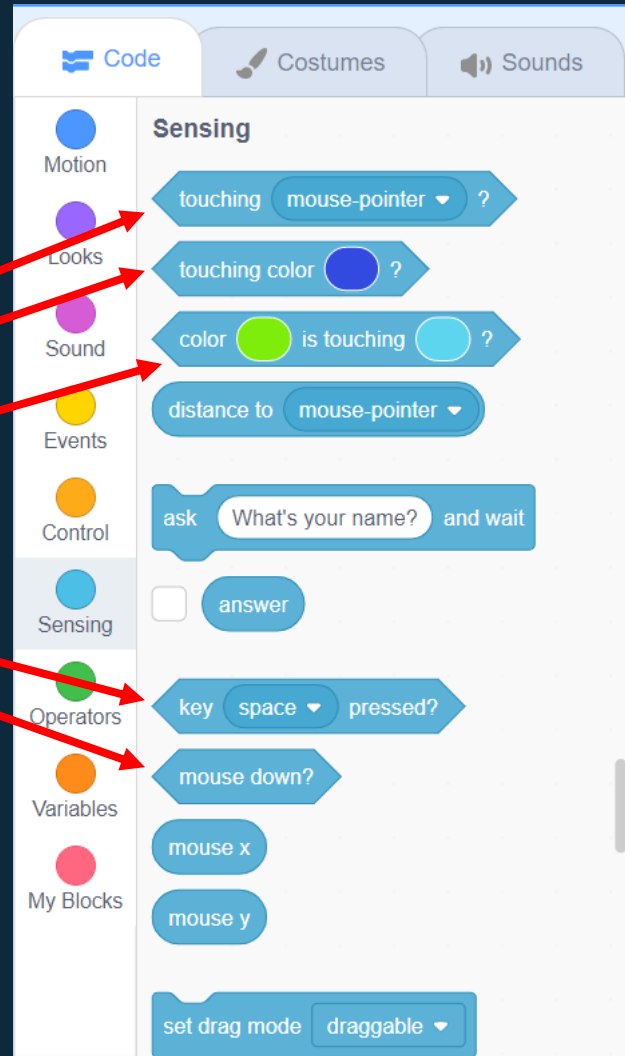
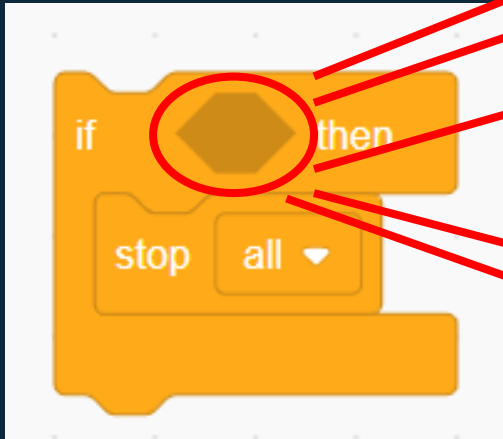
# If Your Sprite Crosses the Finish Line...

Then stop the game!





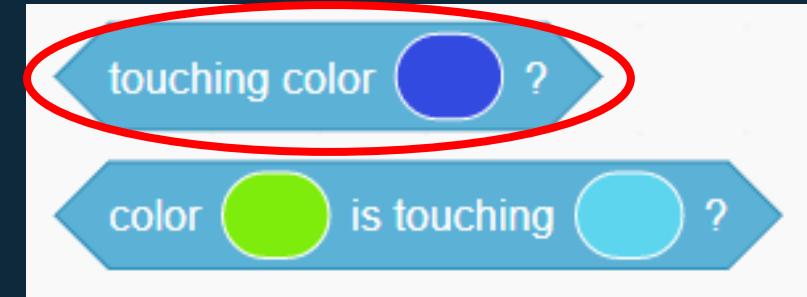
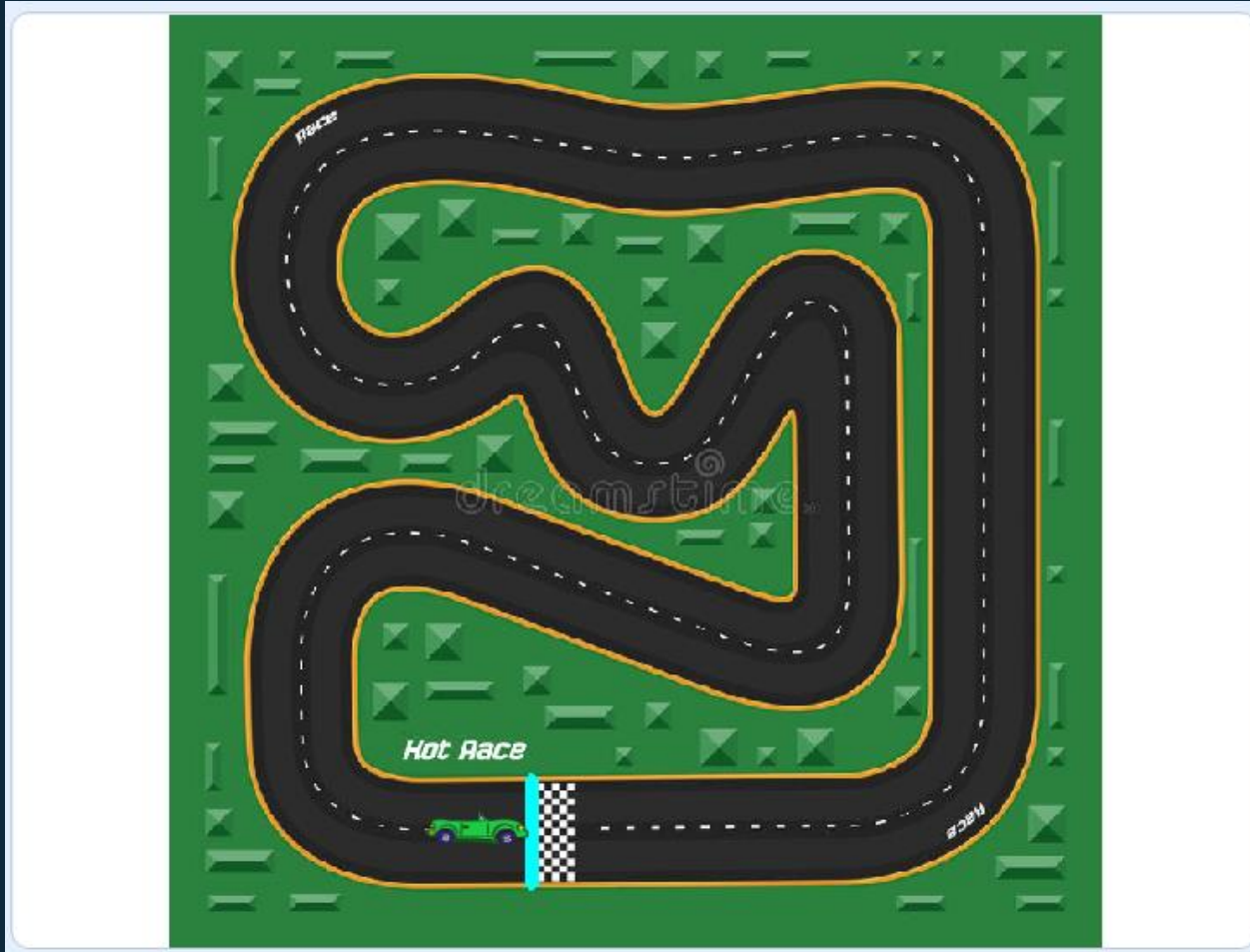
# Shapes Must Match In Scratch




Which condition should we use?



# Touching The Teal Line = Game Finished



Because there is only one moving object (the car) only the top option is needed!



# Where should the if statement go in your code?

In the forever loop under the move action!

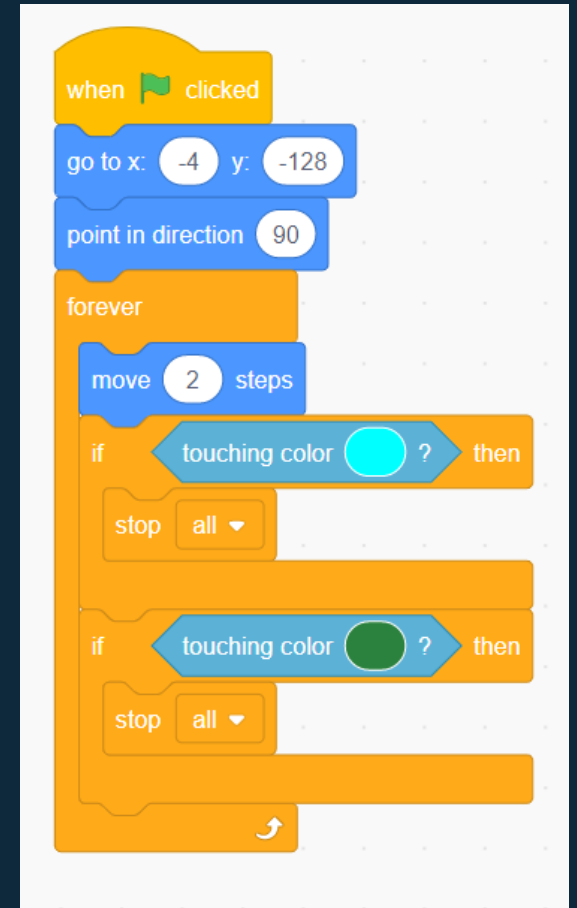
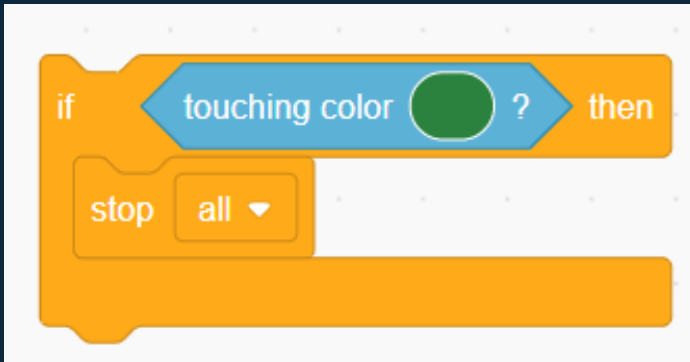
## How should test if the car drives off the road?

A second if statement!

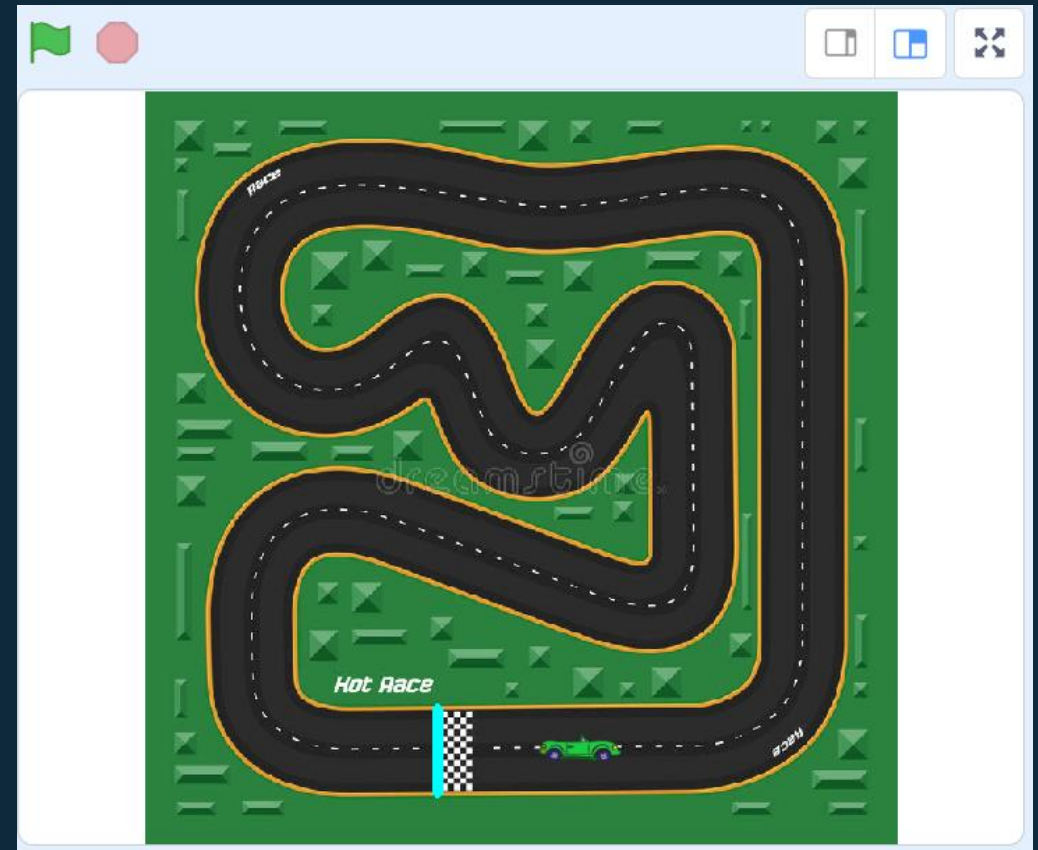
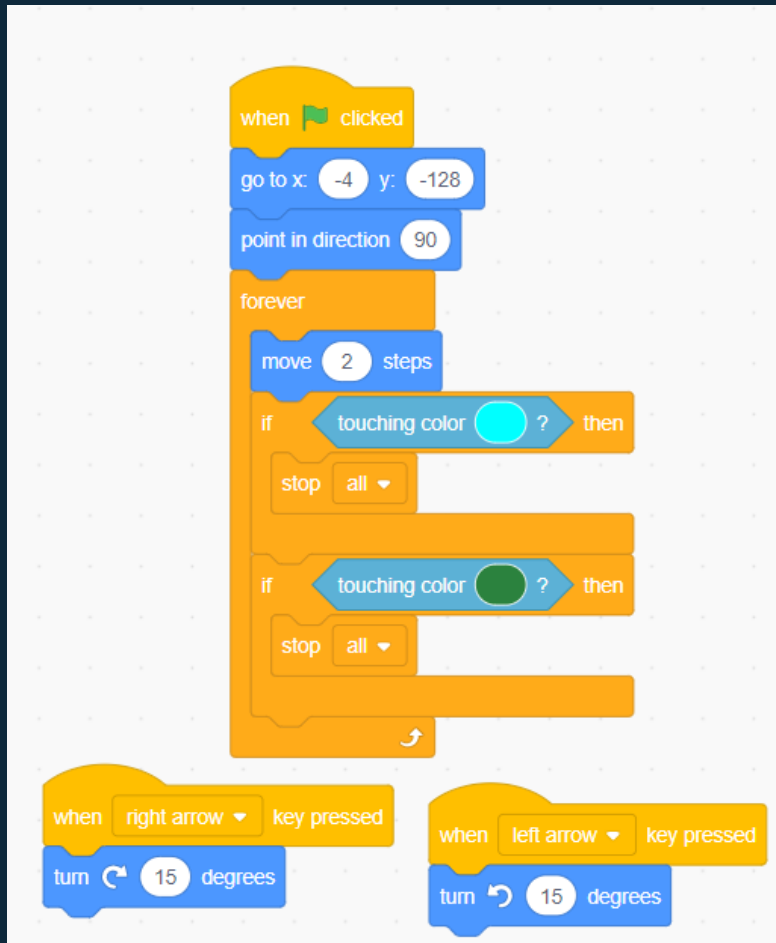
If the car goes off the road,  
**Then** end the game.


What should go inside the if condition?

If it touches the grass color!



# Your Finished Code Should Look Like This





# Congratulations, you completed the game!

If there is time left, let's add some special feature to make your game unique!

## Special Feature Options:

1. [Add power ups to your racetrack](#)
2. [Add sounds after you win or lose](#)
3. [Add a costume to your Sprite after you win or lose](#)

If you're out of time, head over to the [closing questions](#).

A decorative graphic on the left side of the slide. It features a large, solid cyan hexagon in the center. Surrounding it are several smaller hexagons of varying shades of blue and cyan. Some of these hexagons contain white icons: a lightbulb, a thumbs-up, a smartphone, a magnifying glass, a gear, and a speech bubble. There is also a small network diagram icon with a central node and five connecting lines.

# 1. Adding Power Ups

First, we are going to edit the backdrop to add in our power ups.

Then we will write the code that will make your car speed up.

# Choosing the Backdrop

The image displays the Scratch development environment. On the left, the 'Code' tab is active, showing a script for a car sprite. The script begins with a 'when green flag clicked' event, followed by 'go to x: -4 y: -128' and 'point in direction 90'. A 'forever' loop contains a 'move 2 steps' block, an 'if touching color cyan?' block with a 'stop all' block, and another 'if touching color green?' block with a 'stop all' block. Below the loop, there are two 'when arrow key pressed' blocks (right and left) that turn the car 15 degrees. A 'change x by 10' block is at the bottom of the script. The 'Costumes' and 'Sounds' tabs are also visible. The main stage area shows a custom backdrop of a green field with a black winding race track labeled 'Hot Race'. A small green car sprite is positioned at the start of the track. The 'Stage' panel on the right shows the 'Convertible 2' backdrop selected, with its coordinates (x: 28, y: -128) and size (28) displayed. A red circle highlights the 'Backdrops' button in the bottom right corner of the stage panel.

Code

Costumes

Sounds

Motion

move 10 steps

turn 15 degrees

turn 15 degrees

go to random position

go to x: 28 y: -128

glide 1 secs to random position

glide 1 secs to x: 28 y: -128

point in direction 90

point towards mouse-pointer

change x by 10

when green flag clicked

go to x: -4 y: -128

point in direction 90

forever

move 2 steps

if touching color cyan? then

stop all

if touching color green? then

stop all

when right arrow key pressed

turn 15 degrees

when left arrow key pressed

turn 15 degrees

Sprite

Convertible 2

x 28

y -128

Show

Size 28

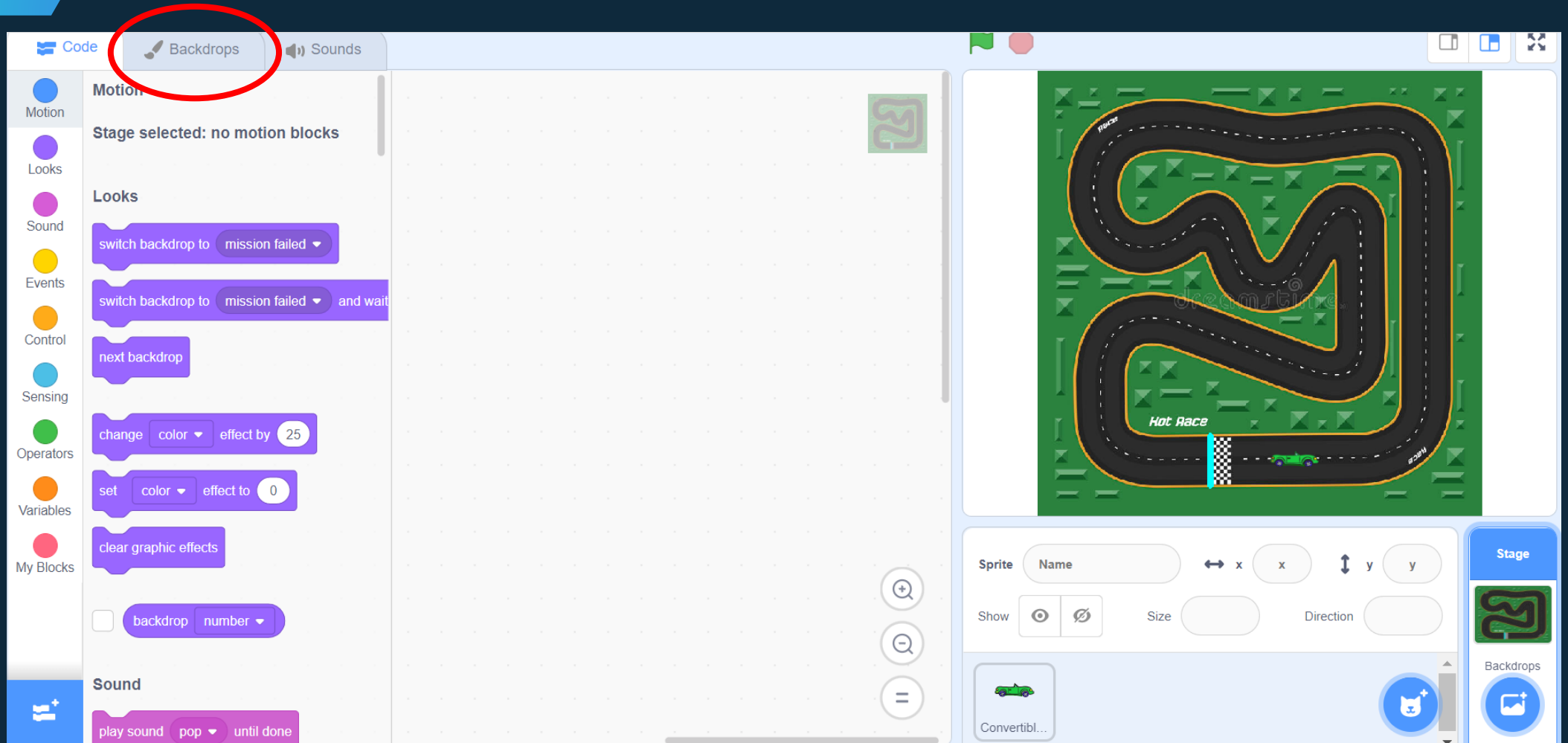
Direction 90

Stage

Backdrops



# Editing the Backdrop



The image shows the Scratch interface with the **Backdrops** tab selected in the top bar. The left sidebar contains the following categories: Motion, Looks, Sound, Events, Control, Sensing, Operators, Variables, and My Blocks. The main workspace is empty. The right sidebar shows the Stage and Backdrops panels. The Stage panel displays a racing track backdrop with a green car and a checkered flag. The Backdrops panel shows a list of backdrops.

**Code** **Backdrops** **Sounds**

**Motion**

Stage selected: no motion blocks

**Looks**

- switch backdrop to mission failed
- switch backdrop to mission failed and wait
- next backdrop
- change color effect by 25
- set color effect to 0
- clear graphic effects
- ☐ backdrop number

**Sound**

- play sound pop until done

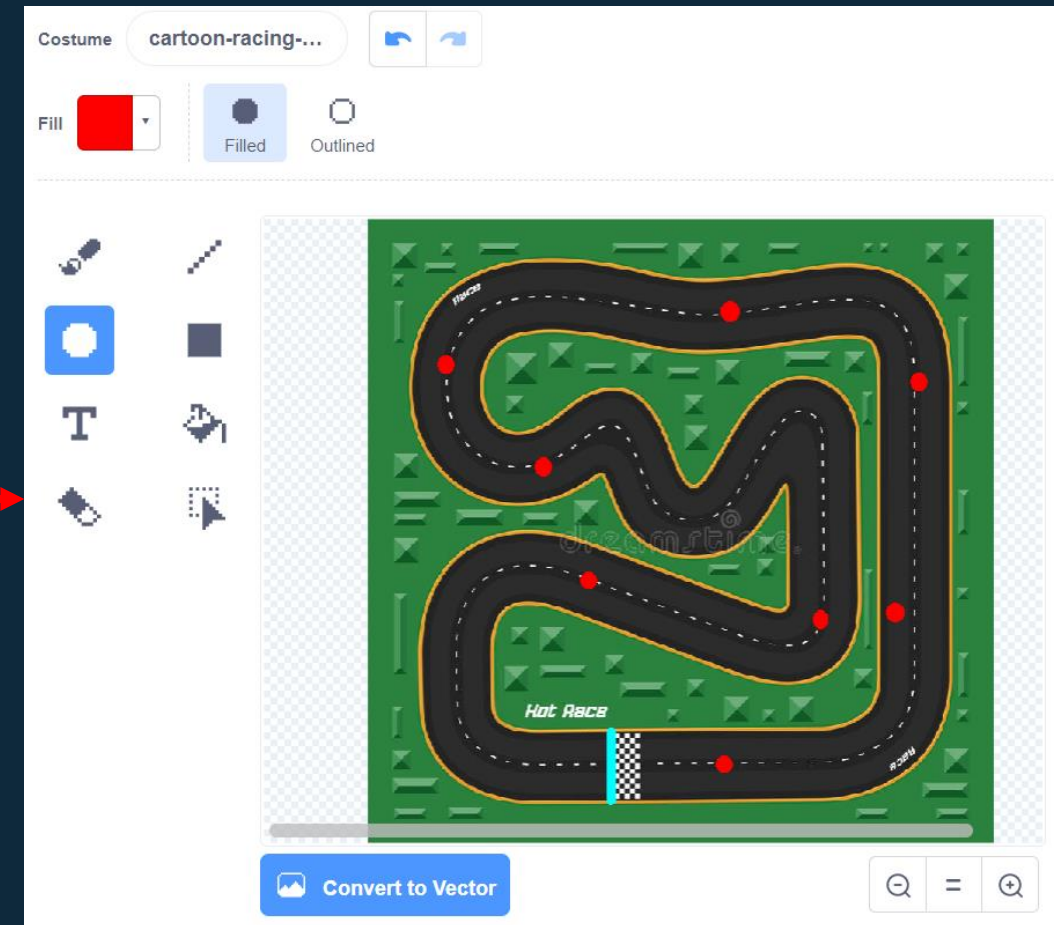
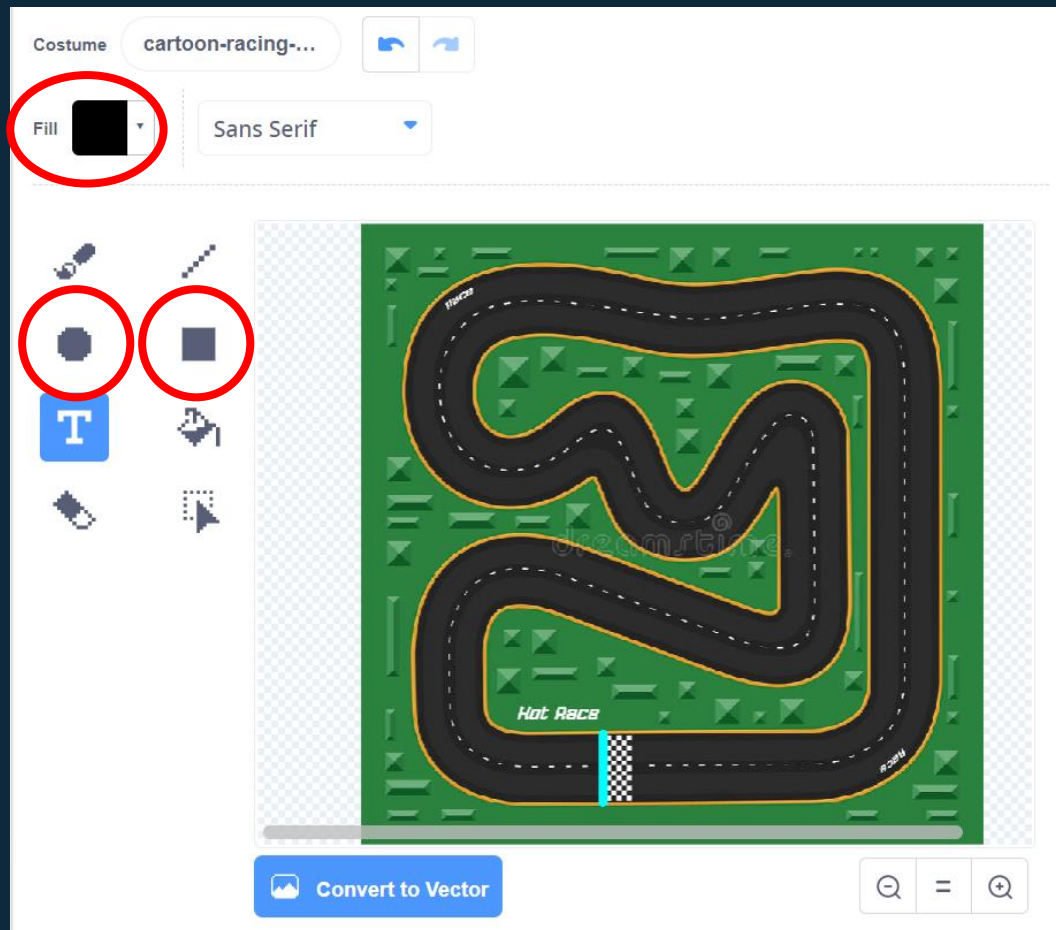
**Stage**

Sprite Name x y Show Size Direction

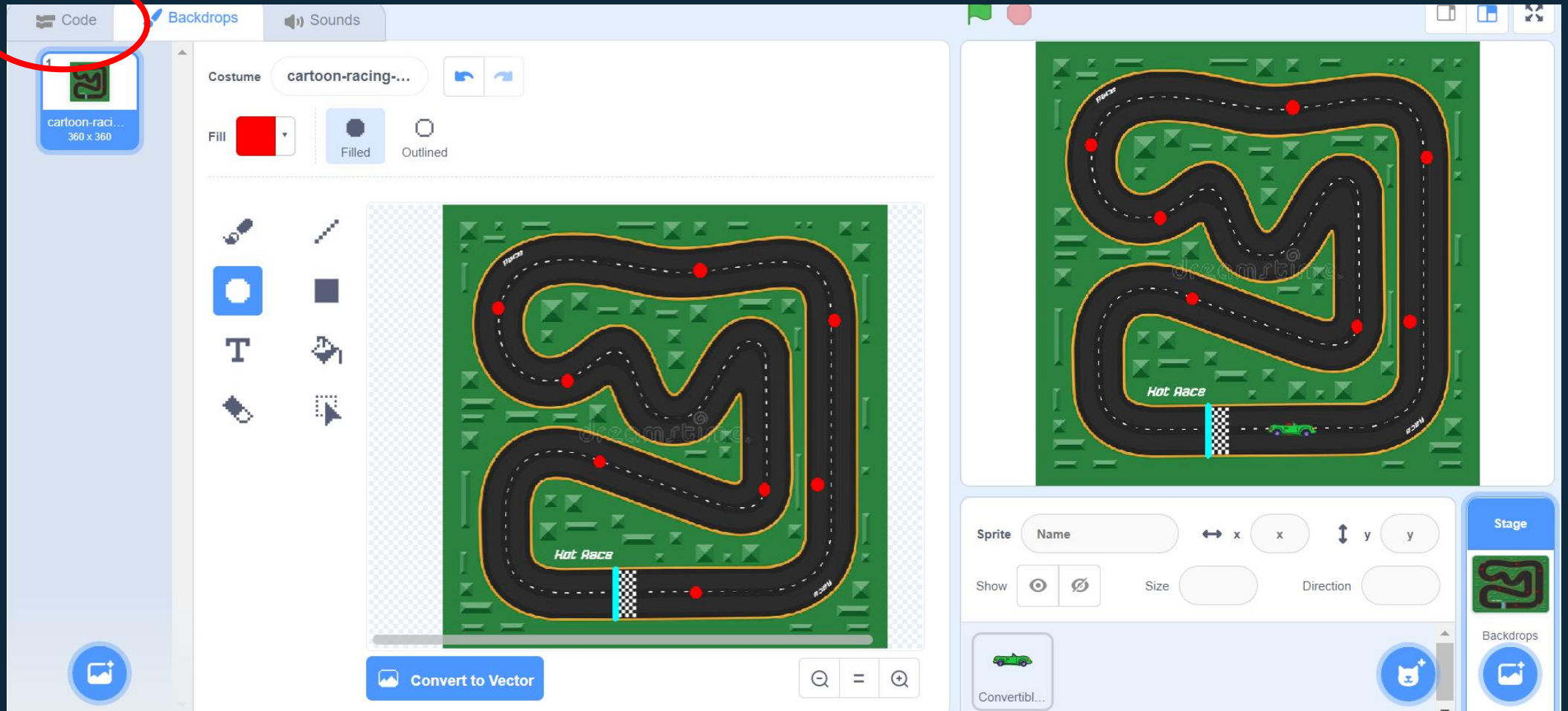
Hot Race

Backdrops

# Placing Power Ups



# Go Back To Code



# Go Back to Sprite

The image shows the Scratch development environment. The left sidebar contains the 'Motion' category selected. The main workspace is empty. The right side shows a stage with a 'cartoon-racing-map-game-4' backdrop. A green car sprite is positioned at the start of the track. The 'Sprite' panel at the bottom right shows the car sprite selected, with a red circle highlighting it. The 'Stage' panel shows the current backdrop and a 'Backdrops' button.

**Code Area:**

- Motion:**
  - Stage selected: no motion blocks
- Looks:**
  - switch backdrop to cartoon-racing-map-game-4
  - switch backdrop to cartoon-racing-map-game-4
  - next backdrop
  - change color effect by 25
  - set color effect to 0
  - clear graphic effects
  - backdrop number
- Sound:**
  - play sound pop until done

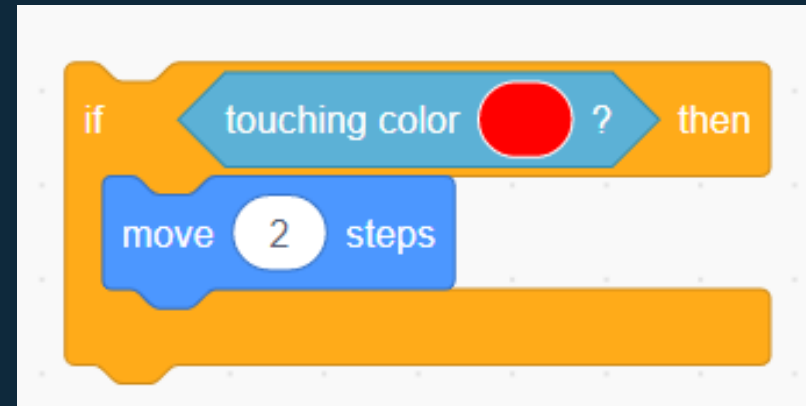
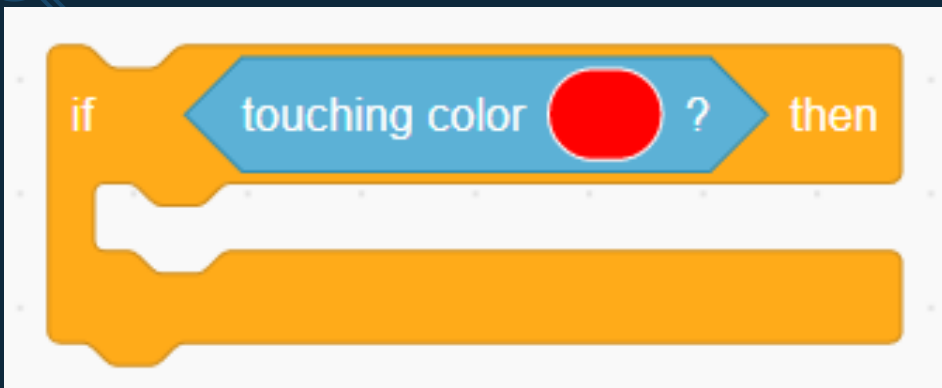
**Stage Area:**

- Backdrop: cartoon-racing-map-game-4
- Sprite: green car (highlighted with a red circle)
- Sprite Name: Convertibl...
- Sprite Size: 100
- Sprite Direction: 90
- Sprite Show: [eye icon]
- Sprite Hide: [eye with slash icon]



# Use An If Statement With Color Sensing

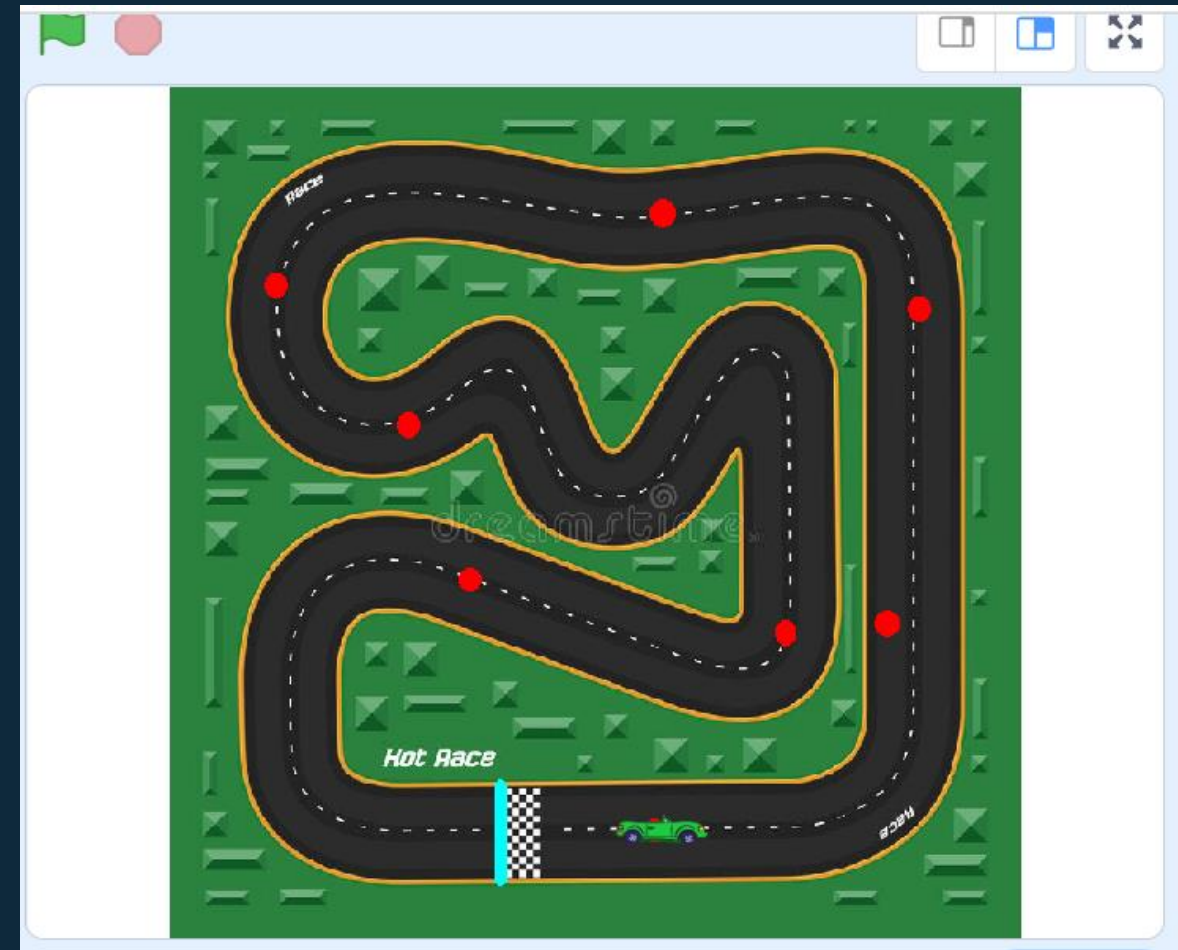
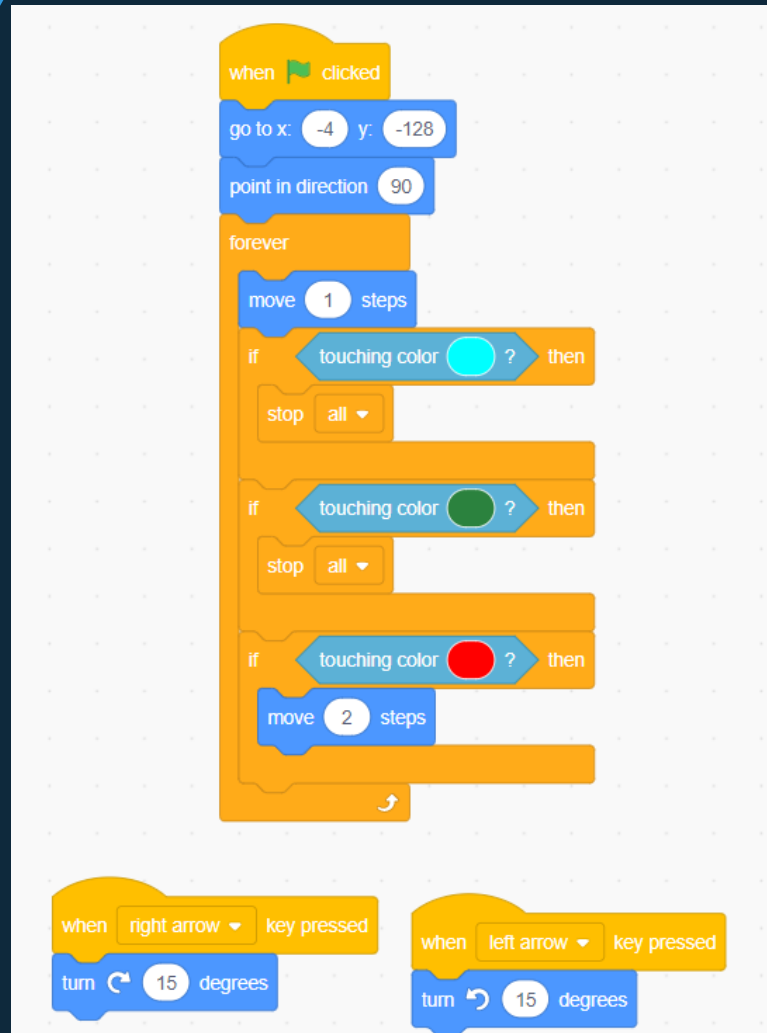
What should go inside?



“Move \_\_ steps”!  
It should be larger than your  
regular movement to be a true  
“power up”. Try different numbers!



# Add To Your Forever Loop





# You added power ups!

Try to play your friends' games!

If you have more time:

1. Add power ups to your racetrack
2. Add sounds after you win or lose
3. Add a costume to your Sprite after you win or lose

If you're out of time, head over to the [closing questions](#).

A decorative graphic on the left side of the slide. It features a large, solid cyan hexagon in the center. Surrounding it are several smaller hexagons of varying shades of blue and cyan. Some of these smaller hexagons contain white icons: a lightbulb, a thumbs-up, a smartphone, a magnifying glass, and a gear. There are also some abstract shapes like a network of dots and a speech bubble.

## 2. Adding Sounds

First, we are going to add the sounds we want to our Sprite.

Then we will write the code that will make your game play music after you win or lose.



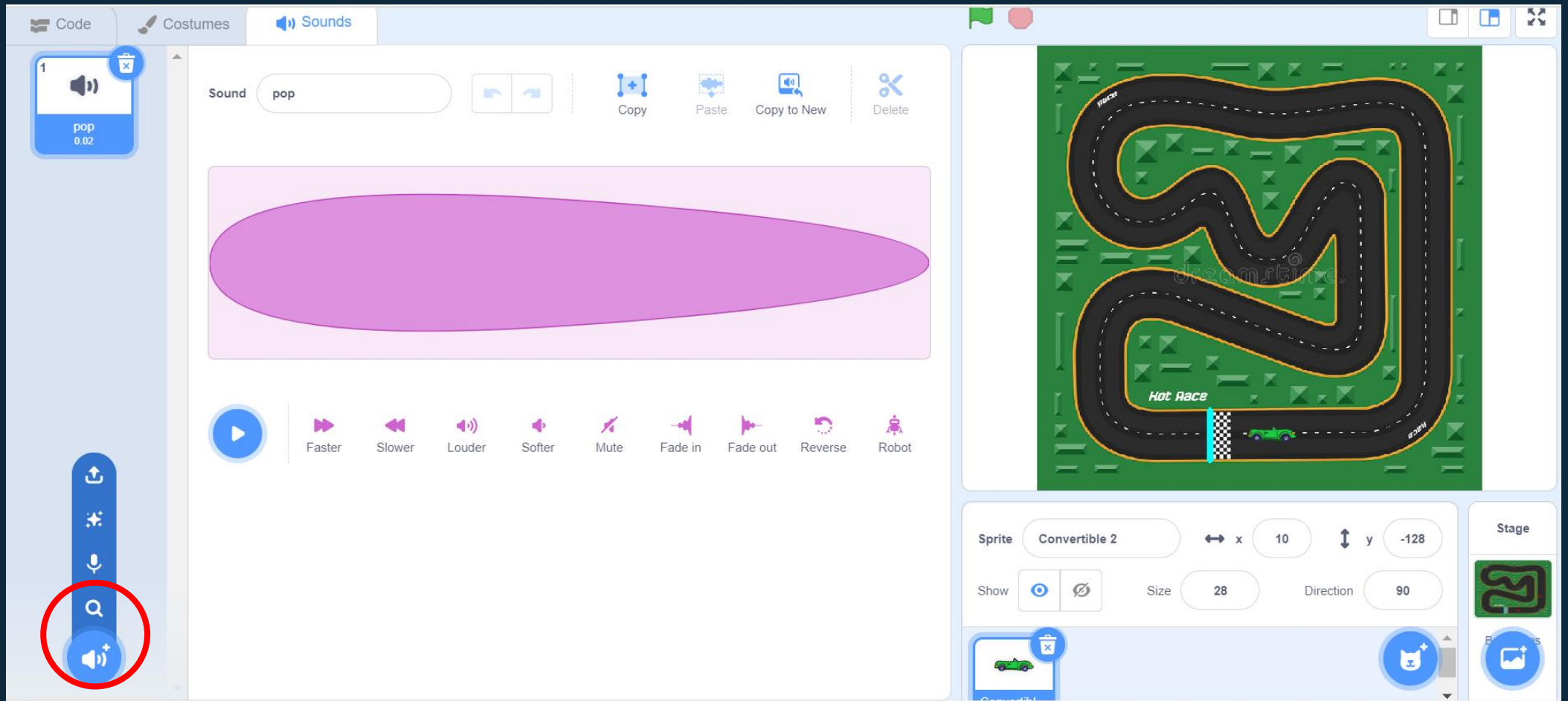
# Finding Sounds

The image shows the Scratch interface for a game titled "Hot Race". The "Sounds" tab is highlighted in the top menu. The code area contains the following blocks:

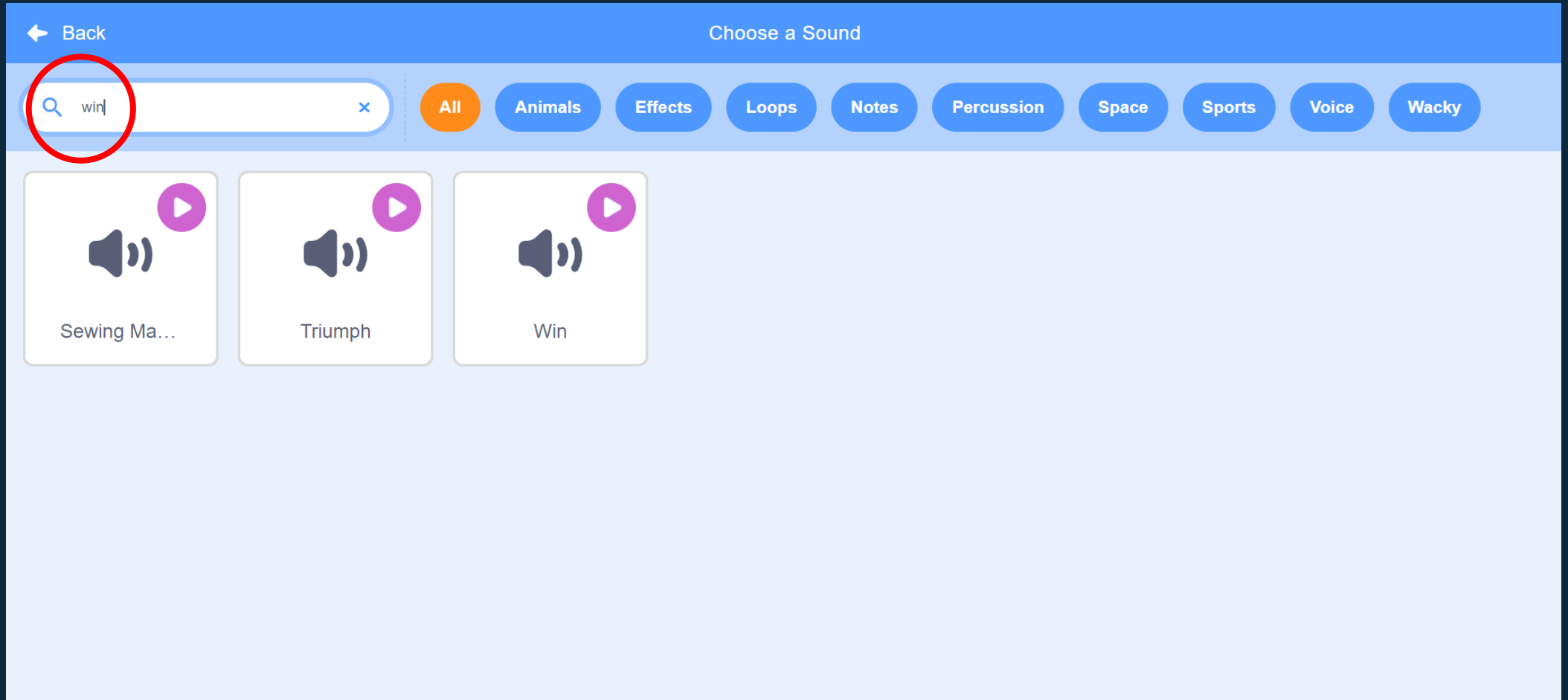
- when clicked
- go to x: -4 y: -128
- point in direction 90
- forever loop:
  - move 2 steps
  - if touching color cyan ? then stop all
  - if touching color green ? then stop all
- when right arrow key pressed turn 15 degrees
- when left arrow key pressed turn 15 degrees

The bottom right shows the sprite "Convertible 2" at x: 28, y: -128, size 28, direction 90.

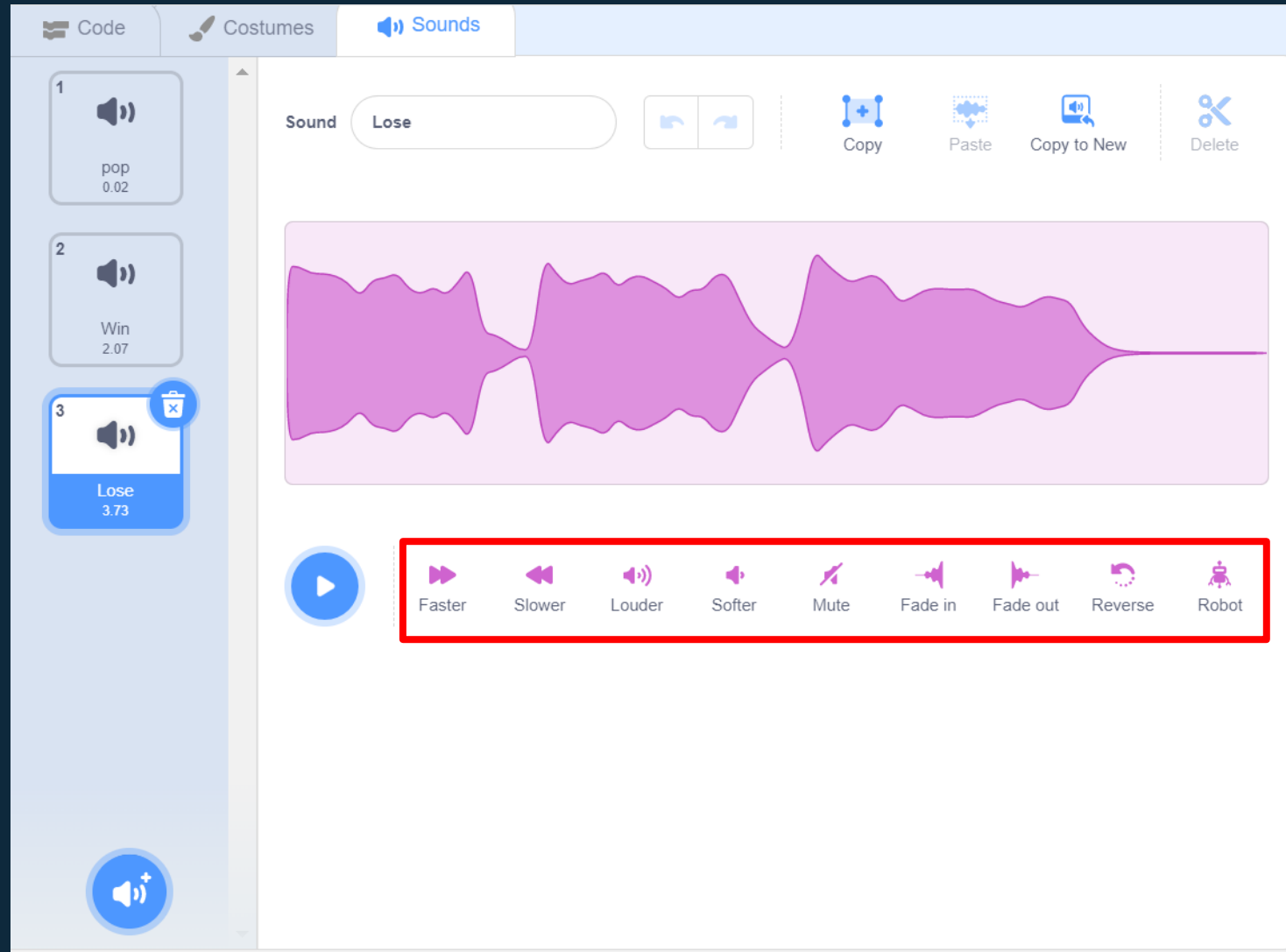
# Search For Sounds



# Choose Sounds For Win and Lose



# Edit Sounds



The image shows the Scratch Sound Editor interface. On the left is a sidebar with a list of sounds: 1. 'pop' (0.02), 2. 'Win' (2.07), and 3. 'Lose' (3.73). The 'Lose' sound is selected and highlighted in blue. The main area shows the selected sound 'Lose' with a pink waveform. Above the waveform are controls for 'Copy', 'Paste', 'Copy to New', and 'Delete'. Below the waveform is a row of editing tools: 'Faster', 'Slower', 'Louder', 'Softer', 'Mute', 'Fade in', 'Fade out', 'Reverse', and 'Robot'. A red rectangle highlights these editing tools. At the bottom left of the main area is a play button icon.

Code Costumes Sounds

1 pop 0.02

2 Win 2.07

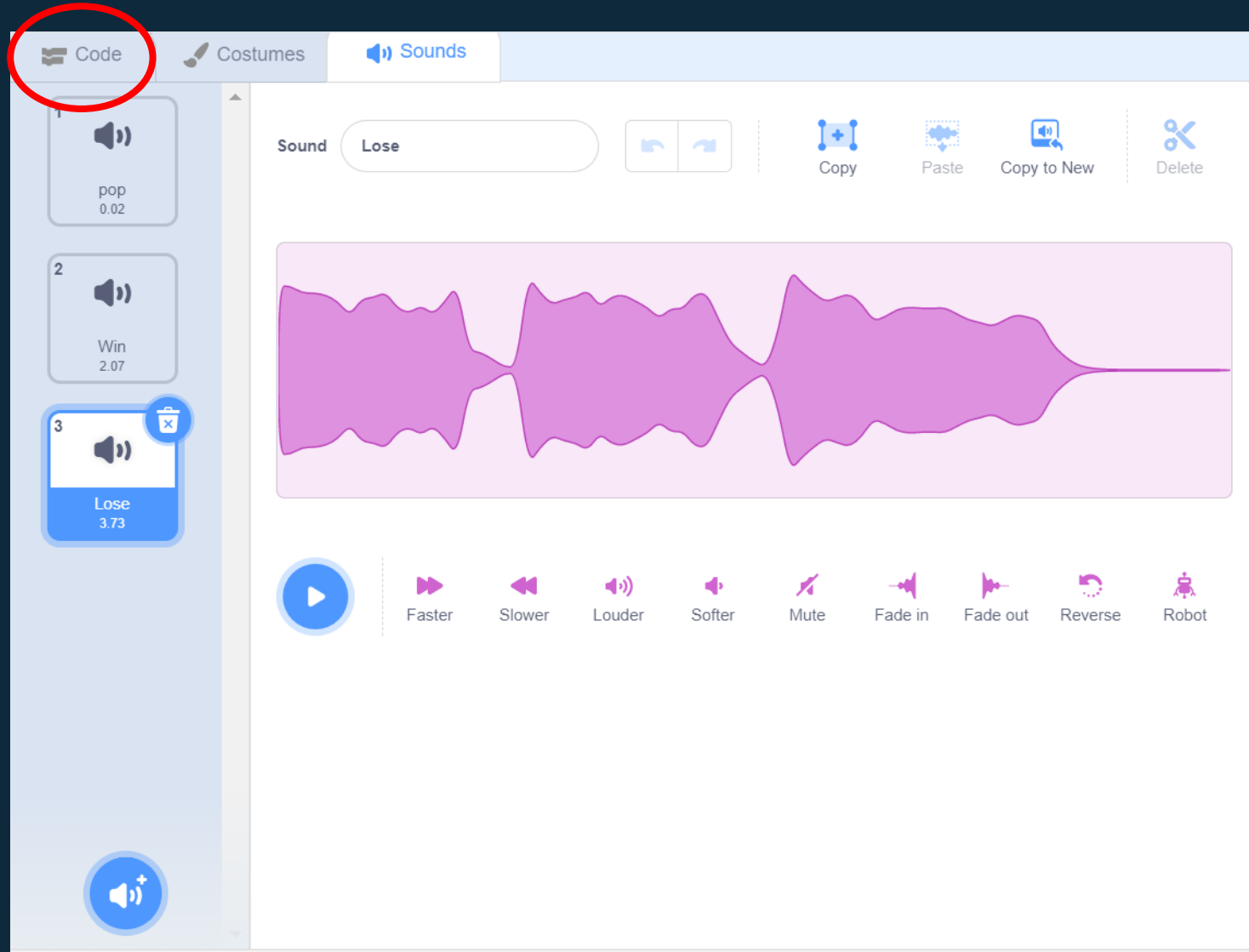
3 Lose 3.73

Sound Lose

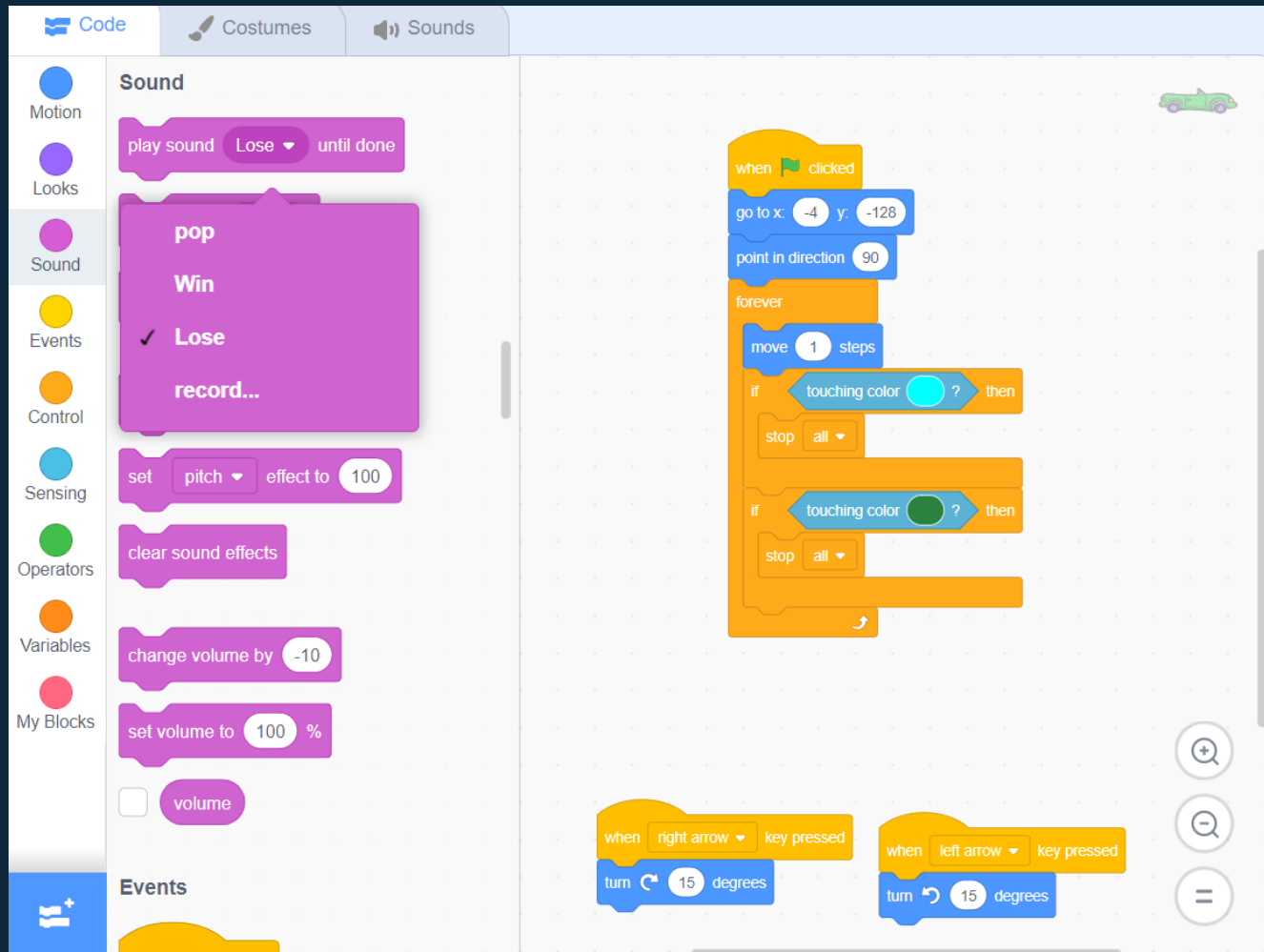
Copy Paste Copy to New Delete

Faster Slower Louder Softer Mute Fade in Fade out Reverse Robot

# Go Back To Code



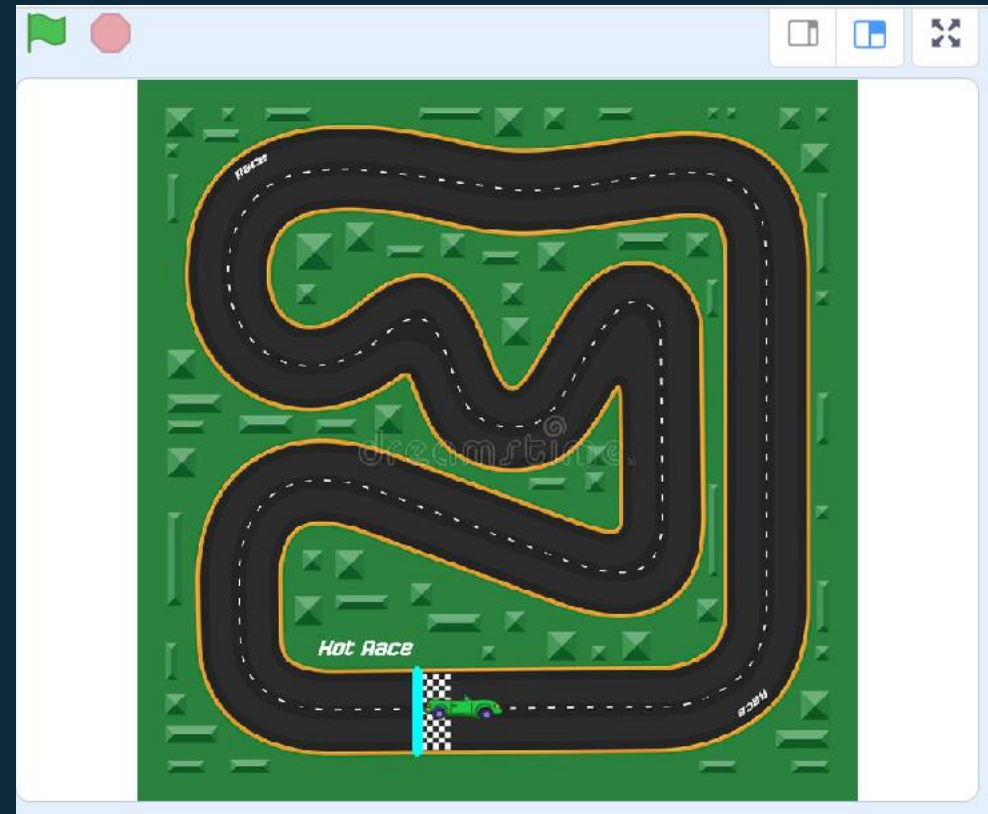
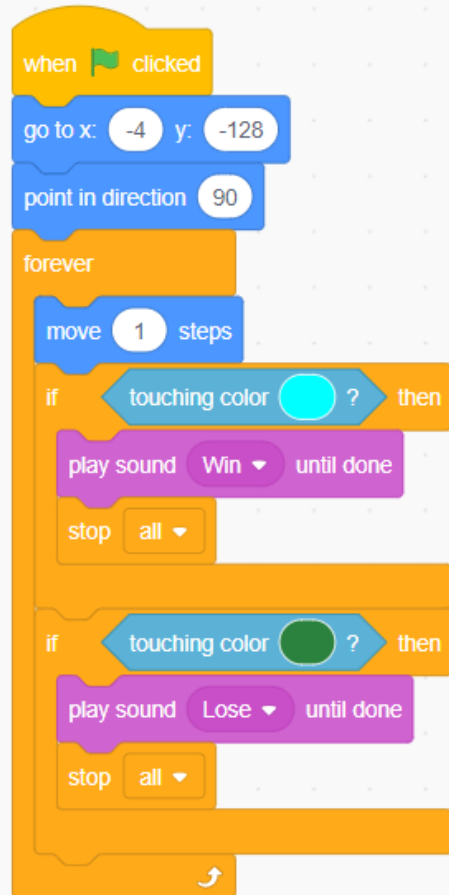
# Find Sound Commands



Where should we add in the sound commands?

# Adding Sounds To Your Code

Add in the appropriate if statement **above** the “stop all” call!





# You added sounds!

Try to play your friends' games!

If you have more time:

1. Add power ups to your racetrack
2. Add sounds after you win or lose
3. Add a costume to your Sprite after you win or lose

If you're out of time, head over to the [closing questions](#).



A decorative graphic on the left side of the slide. It features a large, solid cyan hexagon in the center. Surrounding it are several smaller hexagons of varying shades of blue and cyan. Some of these smaller hexagons contain white icons: a lightbulb, a thumbs-up, a smartphone, a magnifying glass, and a gear. There are also some abstract shapes like a network of dots and a speech bubble.

## 3. Adding Costumes

First, we are going to create new costumes for our Sprite.

Then we will write the code that will make your current Sprite switch when you win or lose.

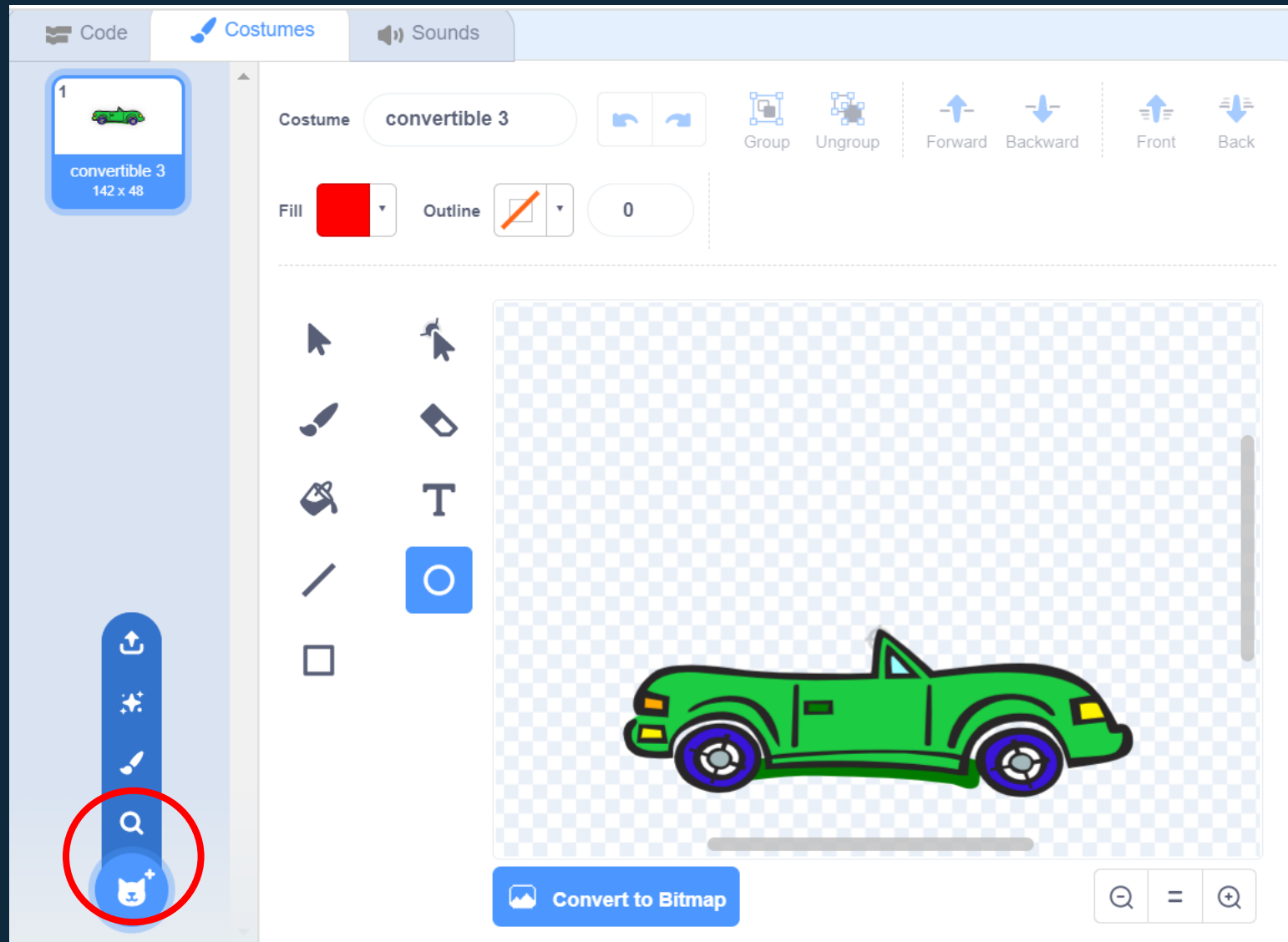
# Finding Costumes

The image shows the Scratch development environment. The 'Costumes' tab is highlighted with a red circle in the top-left pane. The main workspace displays a green car sprite on a racetrack background. The code area contains the following blocks:

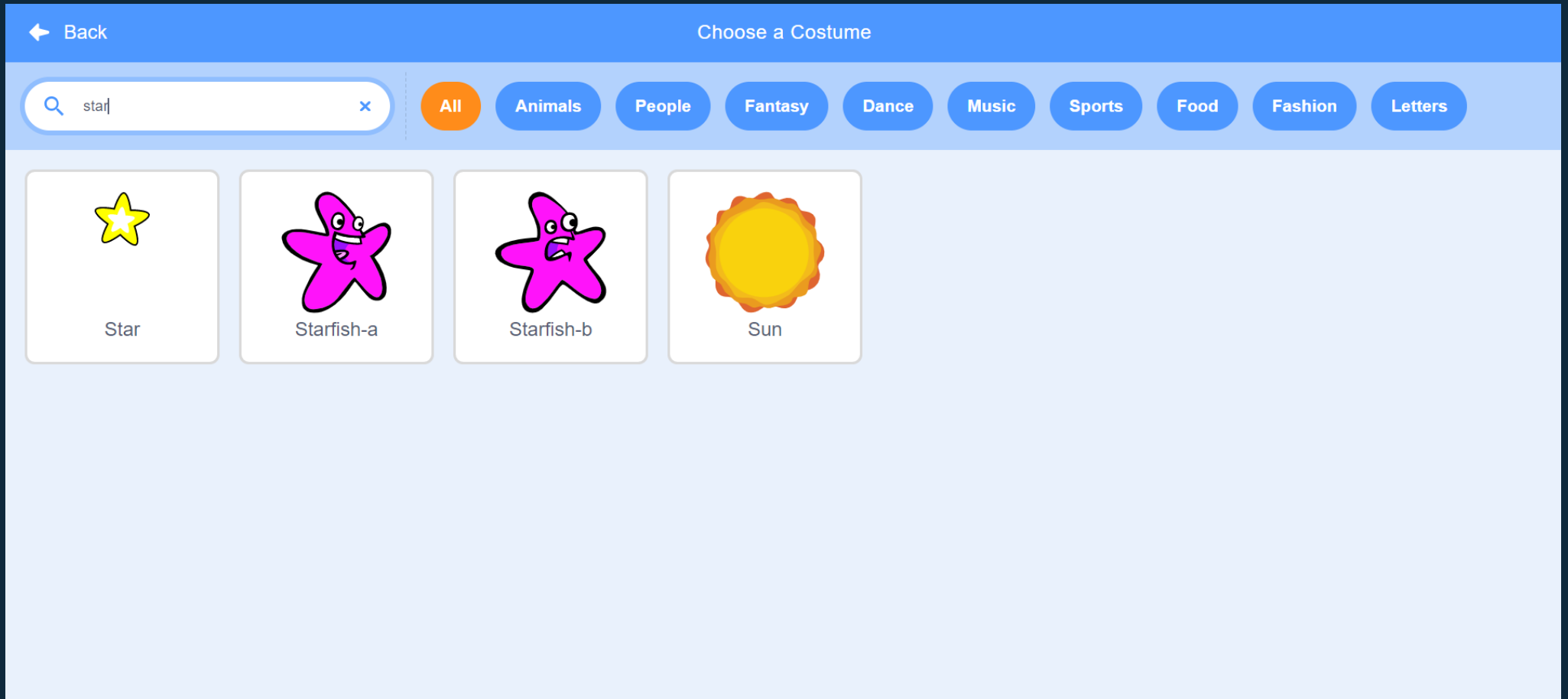
- when green flag clicked**
  - go to x: -4 y: -128
  - point in direction 90
  - forever loop**
    - move 2 steps
    - if touching color cyan? then stop all
    - if touching color green? then stop all
- when right arrow key pressed**
  - turn 15 degrees
- when left arrow key pressed**
  - turn 15 degrees

The bottom-right pane shows the sprite's properties: Sprite 'Convertible 2', x: 28, y: -128, size: 28, direction: 90. The stage shows a racetrack with a green car and a 'Hot Race' text overlay.

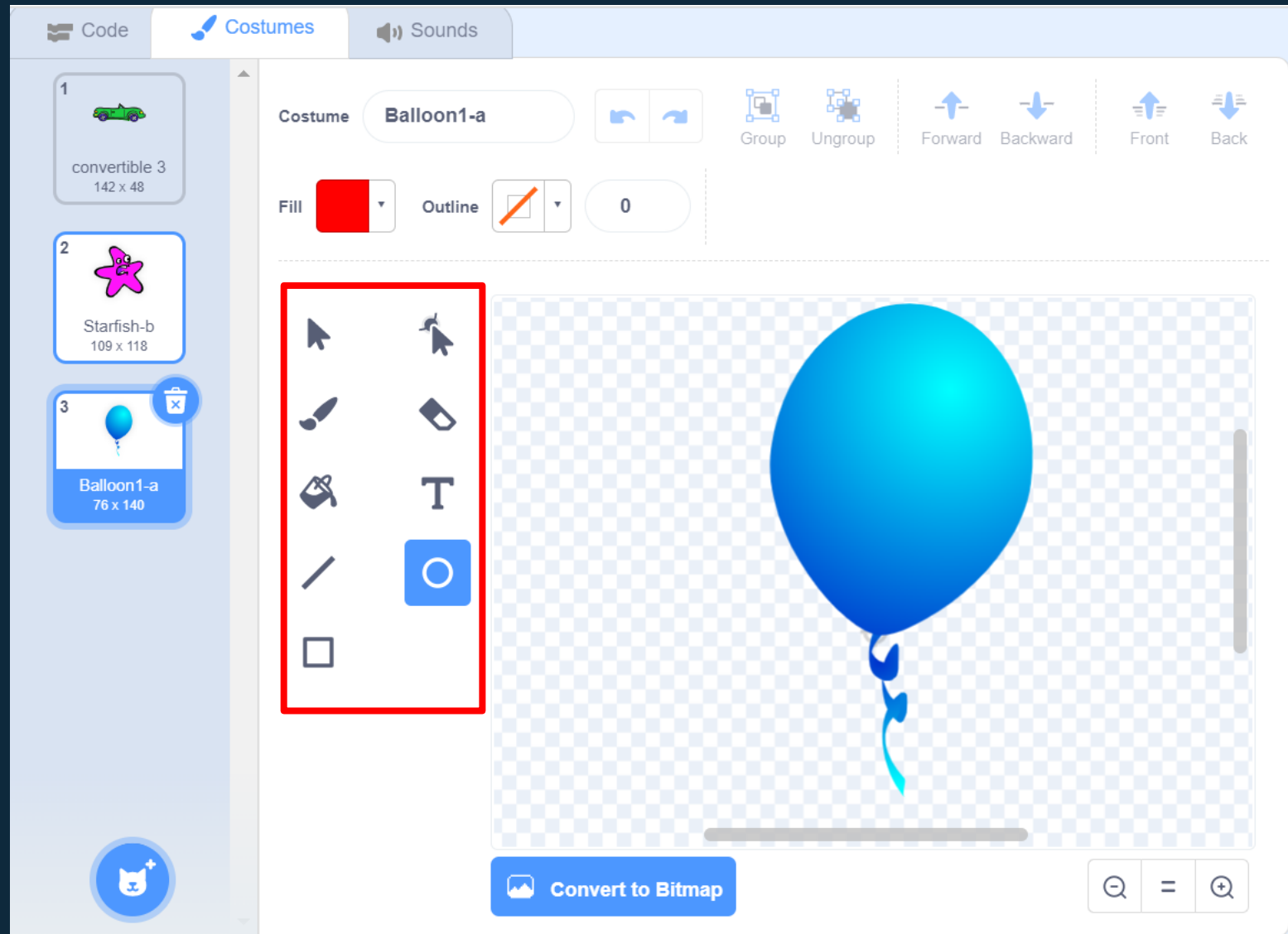
# Search For New Costumes



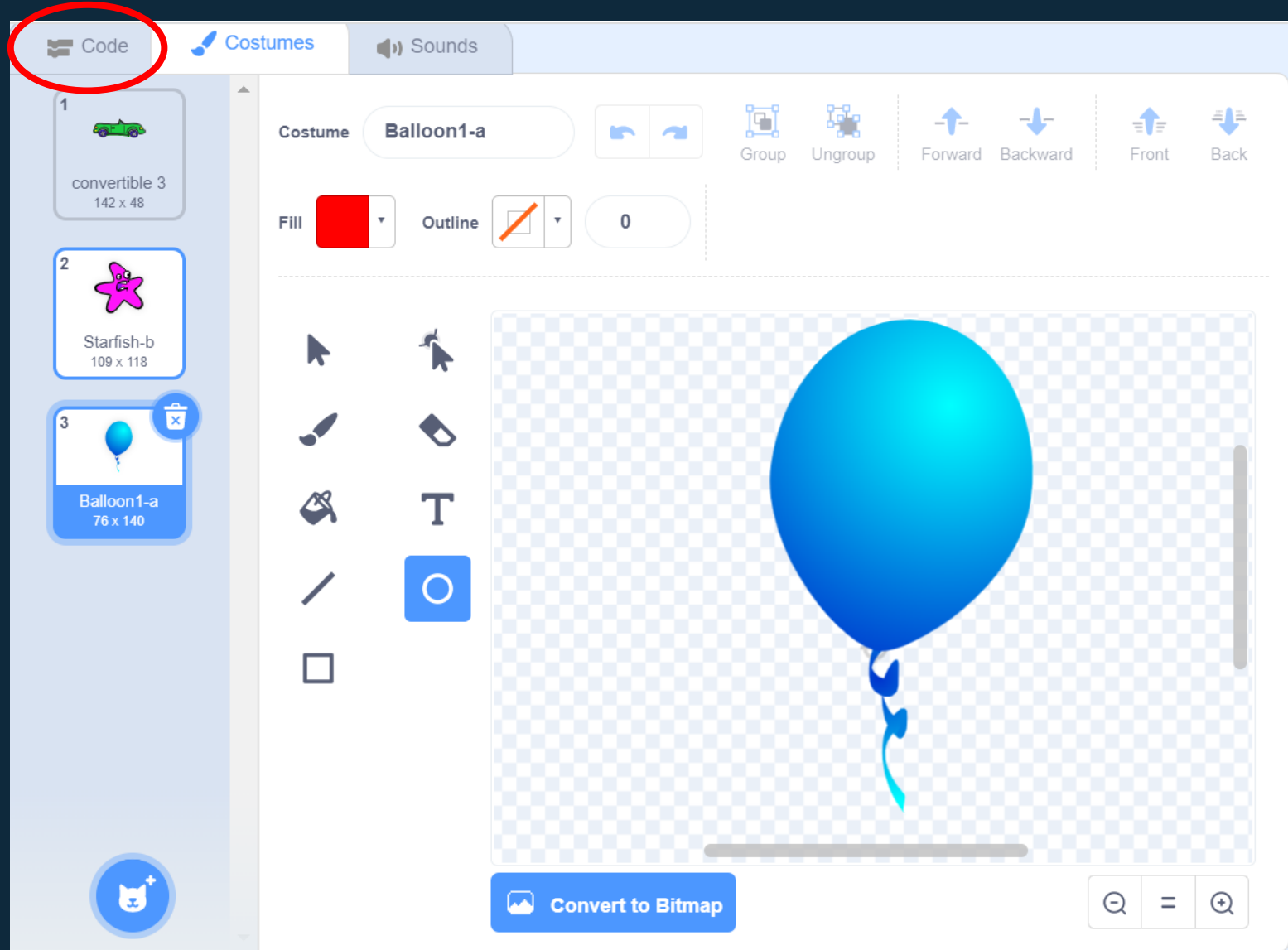
# Choose a Costume For Winning and Losing



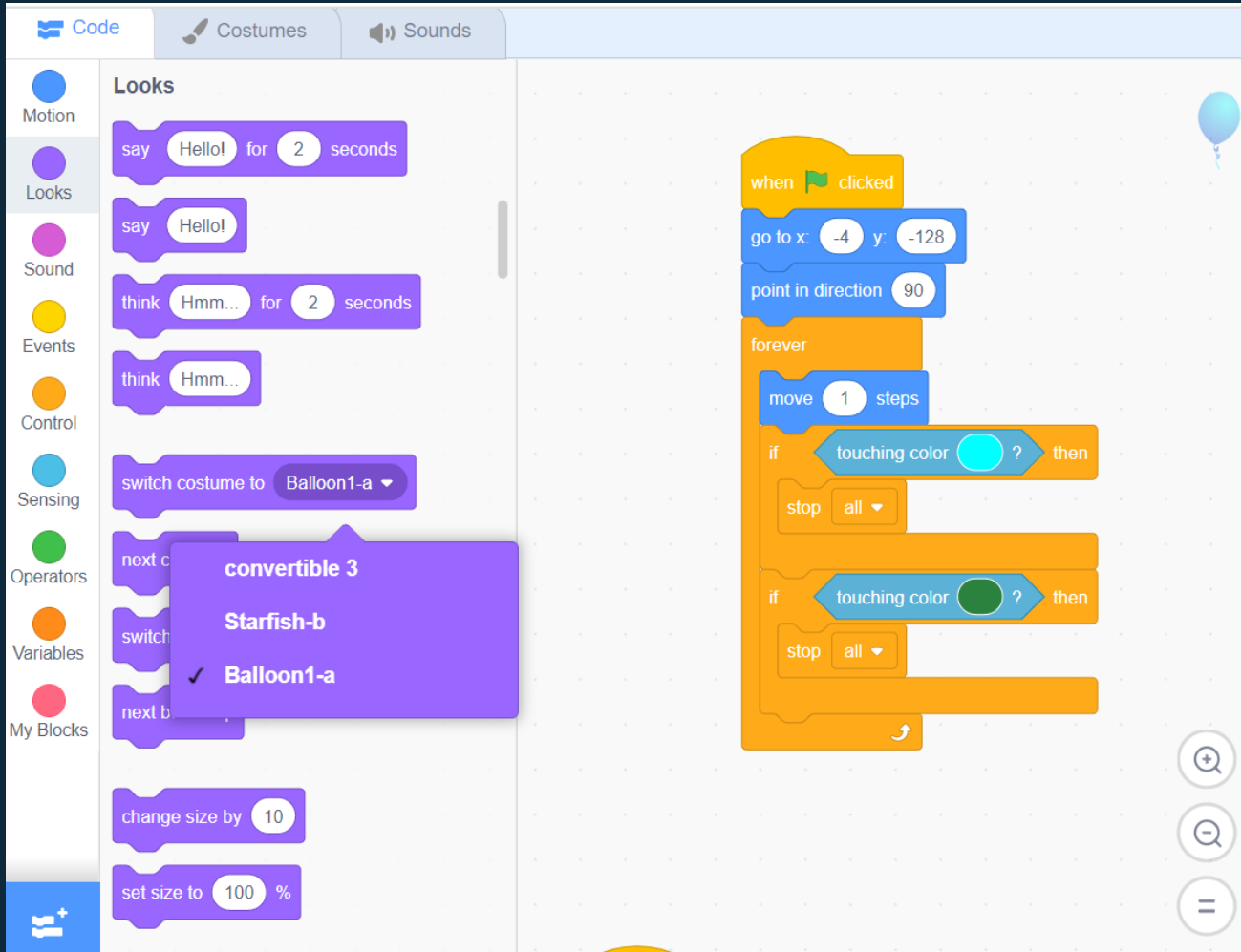
# Edit New Costumes



# Go Back To Code



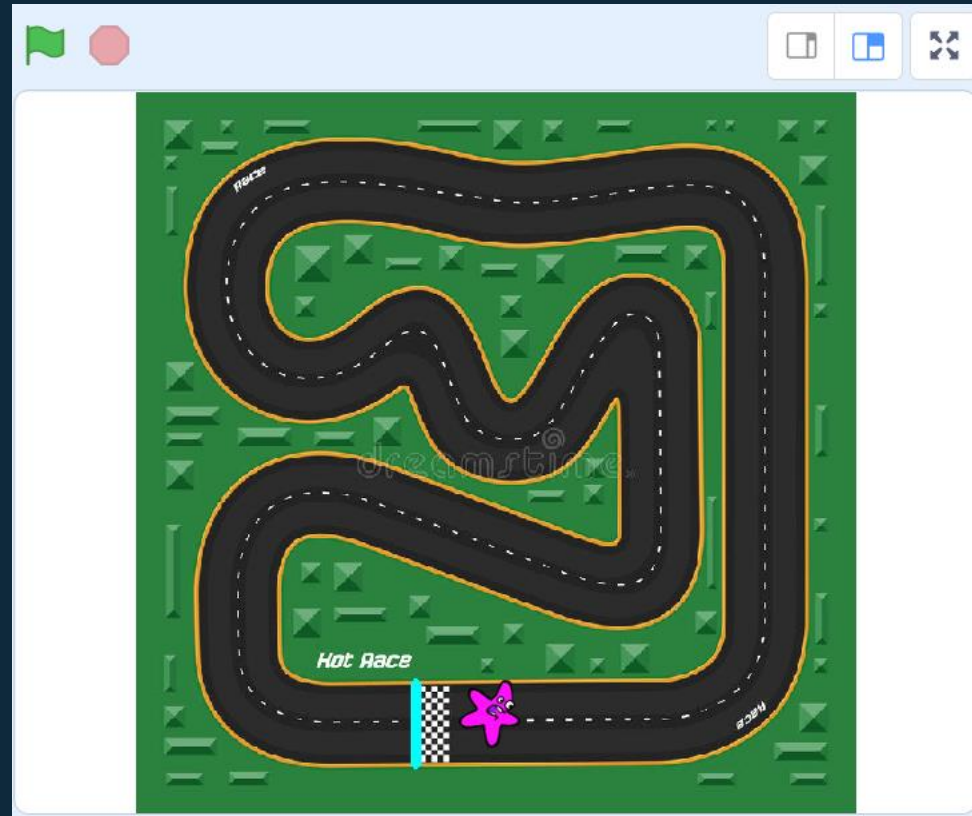
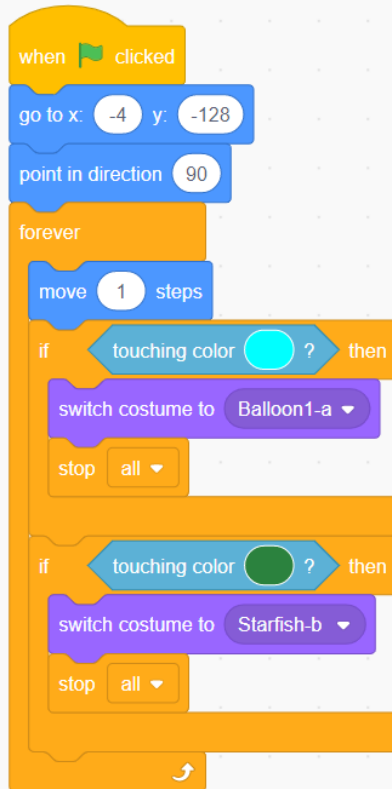
# Find Looks Commands



Where should we add in the sound commands?

# Adding New Costumes To Your Code

Add in the appropriate if statement **above** the “stop all” call!

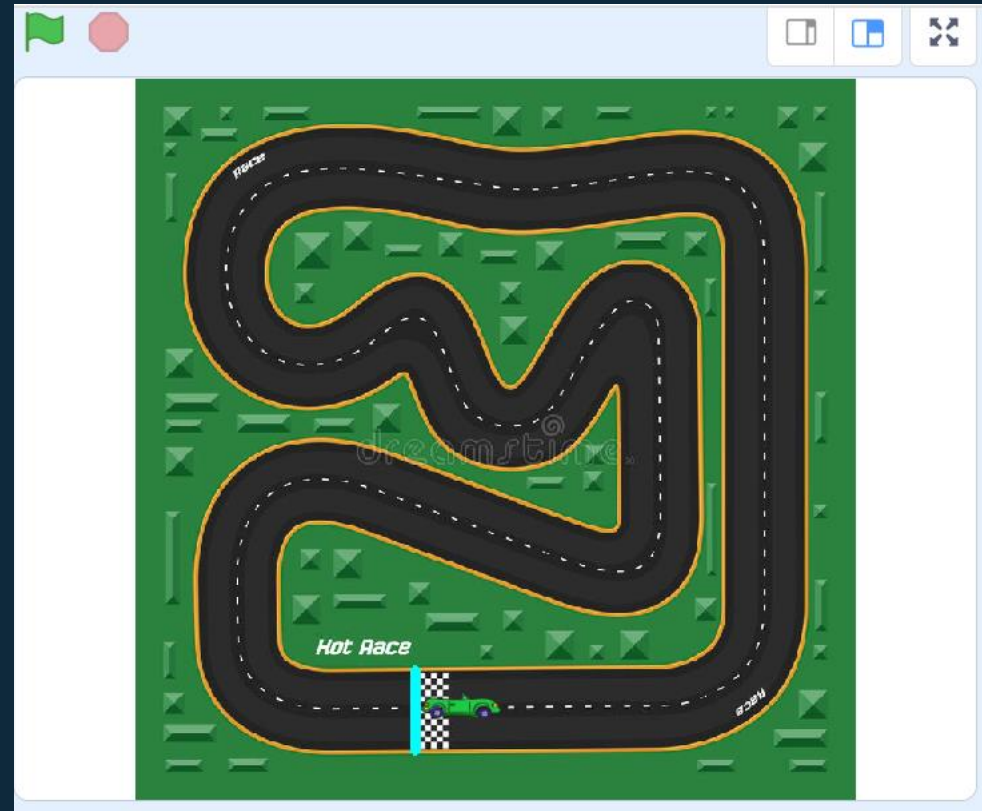
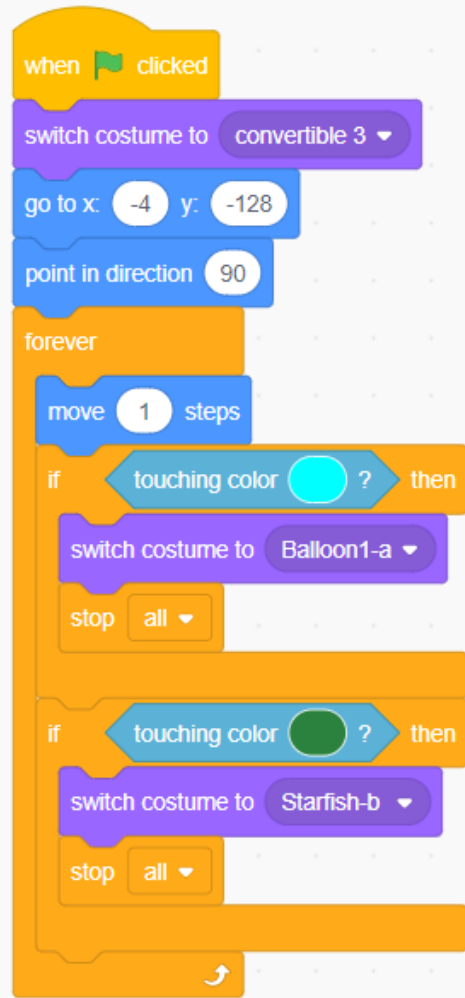


What problem does this create?

How should we fix it?



# Switch Back To Car Costume When the Green Flag Is Clicked






# You added new costumes!

Try to play your friends' games!

If you have more time:

1. Add power ups to your racetrack
2. Add sounds after you win or lose
3. Add a costume to your Sprite after you win or lose

If you're out of time, head over to the [closing questions](#).



# Congratulations, you finished your code!

What part of coding did you enjoy the most?

What part did you enjoy the least?

How can you change the design of your game?

Do you want to try Scratch again?

There are tons of tutorials on how to make your own Scratch video games, movies and more! You can also see and play with projects that others have made at [Scratch.mit.edu](https://scratch.mit.edu)!