

# Buffalo Cart

We all have our favorite racecar games. Buffalo Cart is our version of Nintendo's popular Mario Kart game. We will create a basic racing game where you have to collect coins and avoid obstacles.



**SCRATCH**  
SKILL REQUIREMENTS  
DIFFICULTY 1-5

ELECTRICAL  
PROTOTYPING



ROBOTICS



SOLDERING



PROGRAMMING



DIY



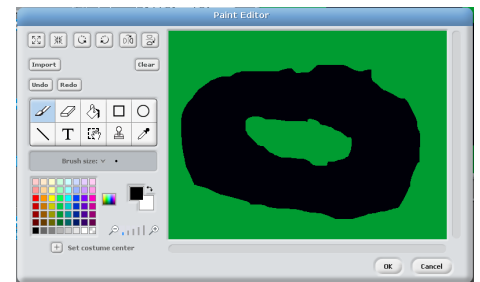
## MATERIALS LIST

- Computer
- Scratch 1.4

### STEP 1: We need a track

The first thing we need to do is create our race track. We accomplish this through creating our own background. At the bottom on the right side, select Stage. Then select the Backgrounds tab between the Scripts and Sounds tab. When you select 'Edit' button, it will open up your editor. By default, the background will be white.

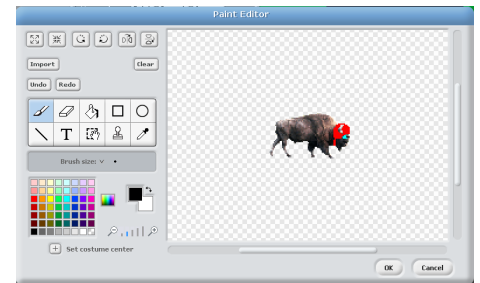
Now your creativity comes into play! Draw a race track that you are going to be racing around. Don't make it too skinny - the idea is to create a donut-like path with an infield and outfield that are the same color. For example, we created a track that is paved (black) and has a grass (green) infield and outfield. You can also add some extra stuff later in the background like grandstands with characters watching the race, a pond, a pits area, or anything else you think your race track needs. Click OK for now!




### STEP 2: Our racer

We now need to add a racer that we can drive around the race track. The first thing we need to do is delete Scratch the cat. At the bottom, right click on Scratch (Sprite1) and select "delete." We can now add the sprite we want to use. We chose the buffalo sprite that comes with Scratch, but you can use whatever sprite you like.

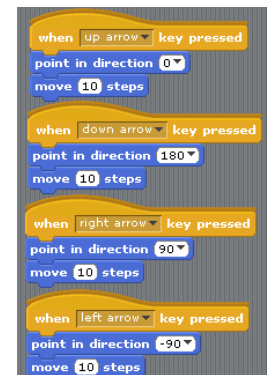
When we import the buffalo racer it ends up being way too big for the race track, so we have to shrink it down. We will do that by editing the sprite, because we also want to add a few things to our buffalo. Click on your new sprite. Click on the



'Costumes' tab for your racer, and click the 'Edit' button on the sprite's costume. You can now use the shrink tool  to make your sprite smaller. You may want to check it on the stage after you shrink it a few times, to make sure that you didn't make it too small. We are also editing the buffalo a bit, to give him a helmet and some racing goggles.

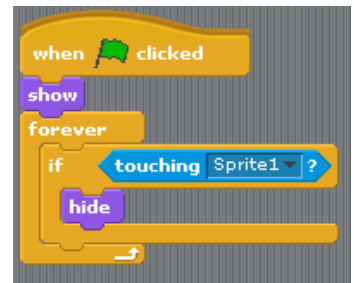
### STEP 3: Driving

We can now use the basic movement setup that we used for Scratch in the Scratcher HotSheet. We are going to click on the 'Scripts' tab and bring in a key-pressed start block. We are then going to change the key to the up arrow key. Now we will add some blocks. We first want our racer to face in the direction of the key (up), so we bring in a 'point in direction' block out of the motion group and we also add a 'move \_\_\_ steps'. You can change the move block number for your racer's speed. Duplicate this group of blocks for each direction. You should have four groups: one for up arrow, down arrow, left arrow and right arrow. Make sure you change the 'point in direction' key to reflect which direction you are going. Try it out! You should be able to drive around on the track!



## STEP 4: Collect the coins!

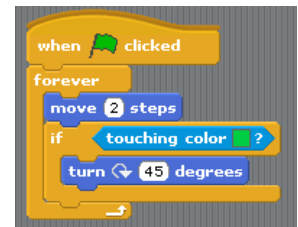
Now we need to create something to drive around and collect. We are going to create a program 1 coin, and then copy it and place duplicates all over the track. We are going to create our coin from scratch! Select 'paint new sprite' to bring up an empty sprite editor and draw a coin to collect. Once you have something drawn, click 'OK' and go to the script tab to start programming.



The goal of this script is for all of the coins to appear at the beginning of the game and disappear when our racer touches them. So, we are going to bring in a green flag block and add a 'show' block from the looks category. This will reset all the coins every time we start the game. We then want to always be checking for contact between the coin and our racer, so we are going to use the 'forever' block. We are now going to add an 'if' block, and for the statement, we are going to select from sensing the 'touching' block and select 'sprite 1'. We want the coin to hide when the 'if' block is true, so we place a 'hide' block inside of the 'if'. You can find 'hide' under the look category. Make sure you test this with a coin. Hit the green flag and collect the coin. If the coin disappears and doesn't reappear, you are good to go. At the bottom right click on your coin sprite and select duplicate to make more coins, and place them around the track for your racer to collect.

## STEP 5: The bad guy

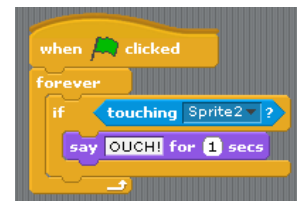
Our last sprite we are going to create is an obstruction, or bad guy of some kind. This bad guy is going to run around the track, and we will lose points if he/she collides with us. We are just going to add a sprite from the file; we are choosing a crab for our game. You may have to shrink your bad guy down to scale before you place it on the track. The script for our bad guy is pretty simple; the goal is for the bad guy to be constantly moving and to bounce off of the sides of the track.



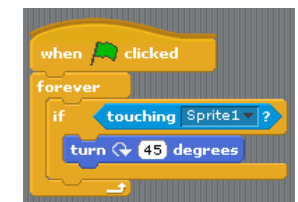
We are going to start with a direction for when green is clicked. When green is clicked, we want the crab to start moving forever and not stop until the end of the game. We first add a 'forever' block, and inside it we add a 'move' block. I changed the number of steps to around three or four. The crab will now move forever! However, we need him to stay on the track and bounce around. We are going to add an 'if' block right after the 'move' block. The argument we are going to add to the 'if' is under the sensing category and its called 'touching color \_\_?'. Add this block to the if statement and click on the color block, and your cursor should turn into an ink dropper; go click on the infield color of your race track. So now any time the crab touches green he will do something, and what we want him to do is turn 45 degrees. This will make the crab change direction when he gets to the edge of the track. Try it out!

## STEP 6: The meetup

We now need to program what happens when the bad guy collides with our racer. At this point, if they run into one another, they just pass by. We first need to decide what will happen to whom when they collide. In our case the racer is going to say, "Ouch!" for one second, and we want the crab to bounce, just like he does when he hits the edge of the track. First we are going to write the script for the racer.



Open up the racer's script and add a 'when green clicked' start block. Under that we want to be checking for the collision between the two. We'll add a 'forever' block with an 'if' block inside it. From the sensing, we grab the first 'touching?' block and place it inside the 'if' block, and we then select the crab sprite from the drop-down menu. Inside the if statement we place a 'say \_\_ for \_\_ seconds' block. We are going to have him say, "Ouch!" for one second.



For the crab, we are going to grab the whole group of blocks from the racer, drag it over to the crab sprite at the bottom, and release. This should copy all of that script to the crab. Now, open up the crab and make a few small changes. The first change is going to be to change the touching pull down value to the racer sprite. The second change is going to be to remove the 'say' block, and replace it with a 'turn' block, similar to it bouncing off of the wall. Try out your game! Think of this as a demo version. In later HotSheets we will revisit this game to add scores and a custom game controller. Enjoy!

- How could you add a second racer for two-player mode?
- Add some grandstands to the background. How could you animate them?