

Scratch in 3 Hours

Hello, my name is Minti Mint! In this handout, I will show you how to create your own computer game!



Bernd Gärtner

ETH

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich



ABZ AUSBILDUNGS- UND BERATUNGSZENTRUM
FÜR INFORMATIKUNTERRICHT

Scratch is a programming language that makes it easy for you to create your own interactive story, game, music, or piece of art, and to share it online as a Scratch project.

In Scratch, you are working with colored blocks that you are stacking together like toy bricks to create your program!

For more information, and to program with Scratch, go to <http://scratch.mit.edu/>!

In this handout, we're programming a game together in which the goal is to steer Scratch cat across a race track.

Before that, I recommend you to look at my accompanying online tutorial „Scratch in 5 Minutes“, or to have a person already familiar with Scratch showing you once how the racing game is made.

After working through this handout, you are ready to create your own games!



How to use
this handout:

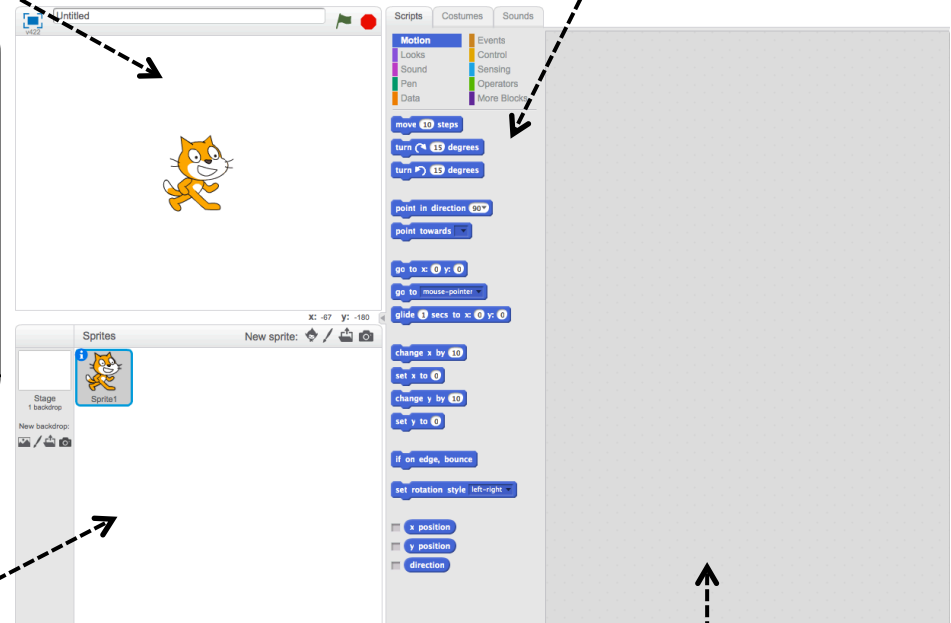


The accompanying online tutorial „Scratch in 5 Minutes“ can be found on the Scratch website at <http://scratch.mit.edu/projects/23844258/>

Stage: This is where your *project* runs. The acting creatures or things in your project are called *sprites*. The project contains your program but also all costumes, sounds, and backdrops.

Block palette: From here, you are dragging command blocks into the programming area in order to build your program.

This is how your screen looks like after clicking „Create“ on the Scratch website. There are four important areas I need to explain to you.



Sprite list: Here you choose whom you are currently programming (some sprite, or the stage). In the beginning, Scratch cat is the only sprite.

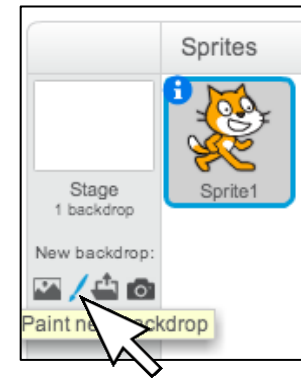
Programming area: Here, you are stacking blocks and build *scripts* to tell the sprites and the stage what to do. All scripts together form the *program*.



Let's get started!
Put your race track on stage!



1. In the sprite list, click on the pen symbol under „New backdrop“



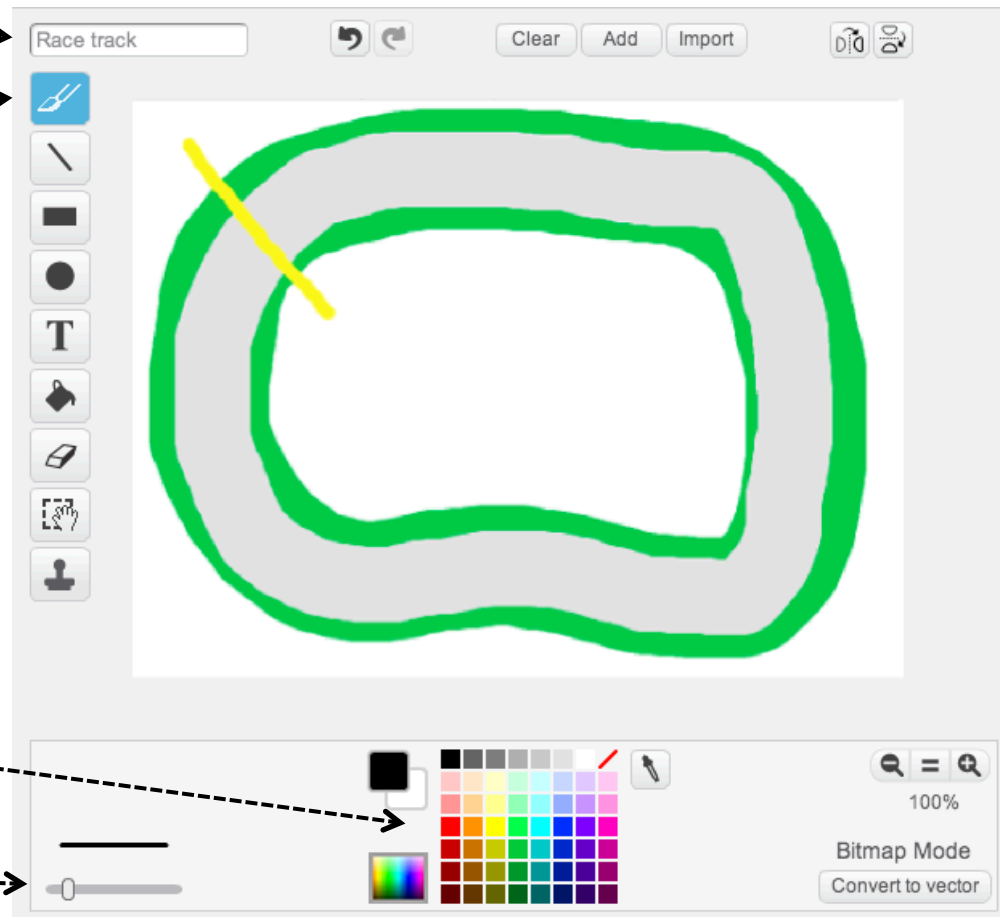
2. Paint the race track, with start/finish line!

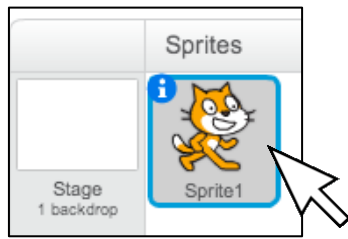
Backdrop name
(optional)

brush

brush
color

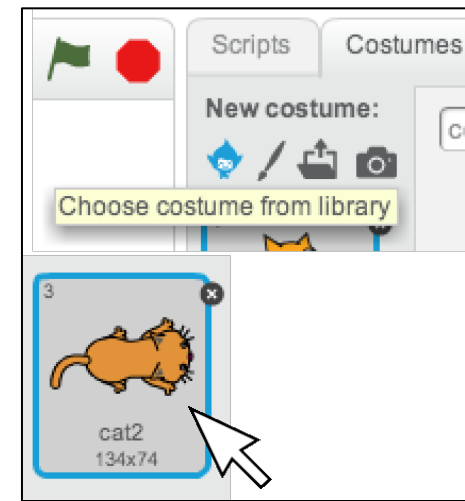
brush
size



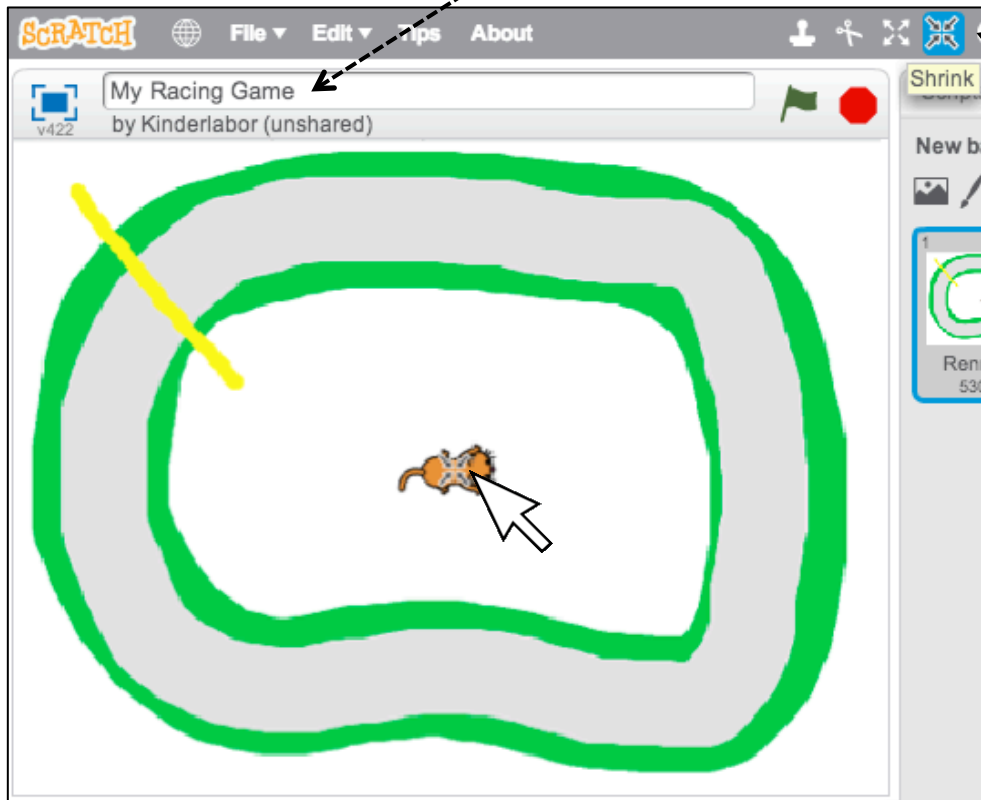


1. Click on Scratch cat in the sprite list!

2. In the programming area, change to **Costumes**, click on the sprite symbol below „New costume“ and select the **cat2** costume from the library!



3. Press the shrink button and click on Scratch cat on the stage! You should also give your project a good name here!



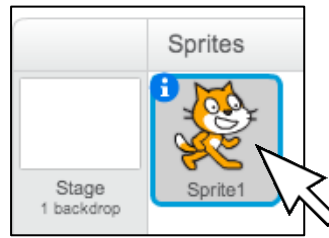
Shrink button

Next turn Scratch cat into a racing cat!

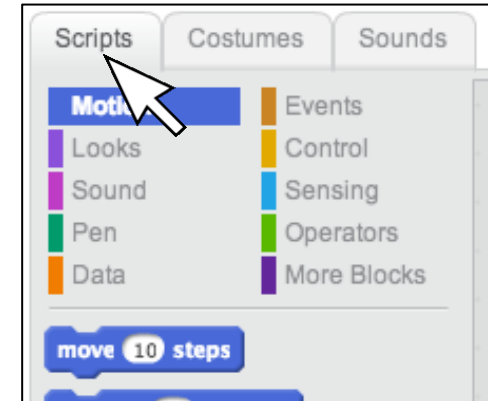


Make sure that Scratch cat fits well on the grey part of the race track!

Now for the actual programming! Here is how you do it!



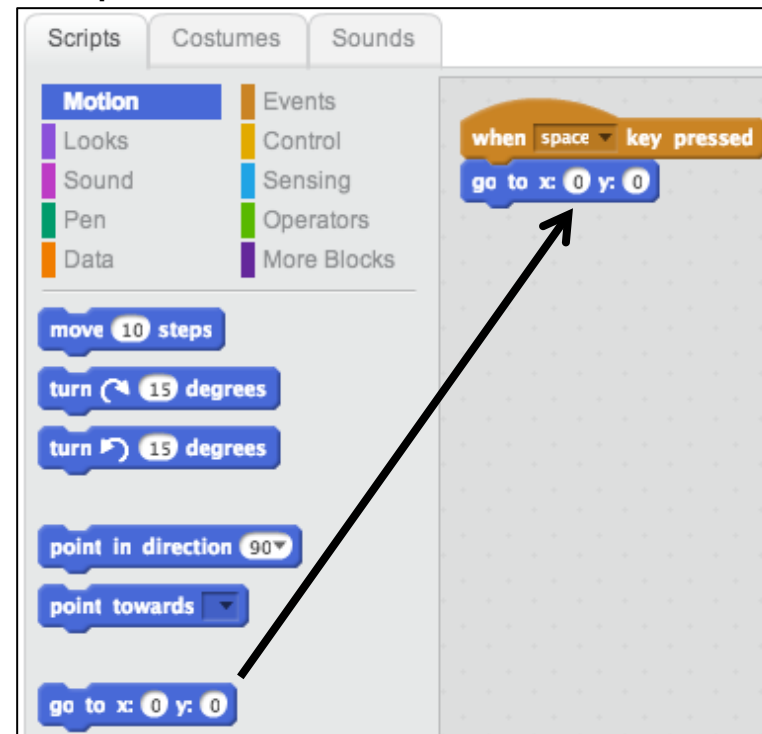
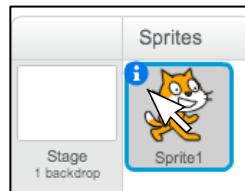
1. Click on Scratch cat in the sprite list!



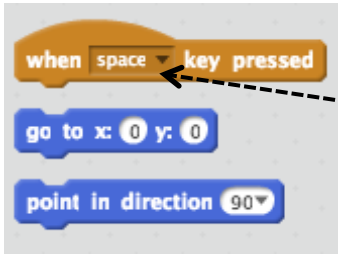
2. In the programming area, change to **Scripts**!

3. Drag commands from the block palette into the programming area and stack them together to form scripts!

Hint: By clicking on the „i“, you can give Scratch cat a real name!



Use these commands:



Select the ,s' key from the menu!

Program Scratch cat such that she goes to the start when the ,s' key is pressed (,s' for start!). For this, you need two devices: mouse pointer and compass !

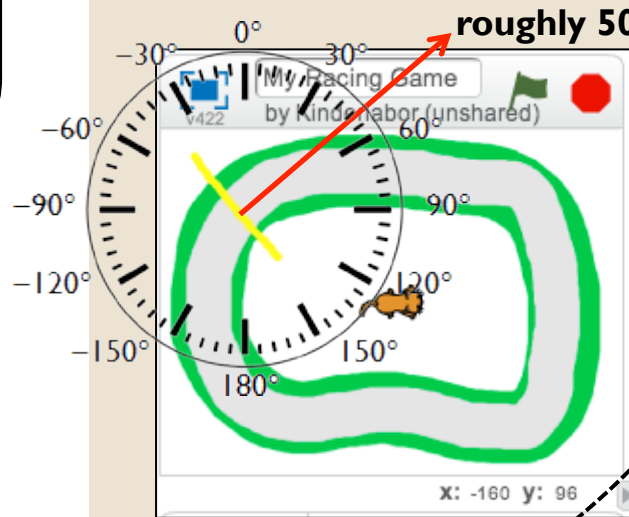


Choose the position:



This means: 160 steps (= pixels) to the left of the center, and 96 steps above the center of the stage.

Choose the direction:



This means: Direction 50 Degrees!

Hover with the mouse pointer above the start line, read off x and y, and put them into the go-to command!

Imagine the start direction as a red arrow, and estimate the direction using the compass. Put the estimated value into the point-in-direction command!

Use these commands:



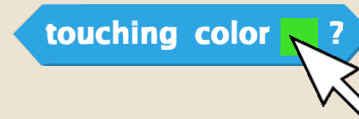
Extend your script to a racing script! Read under **Process** what Scratch cat is supposed to do, and give her the corresponding commands!



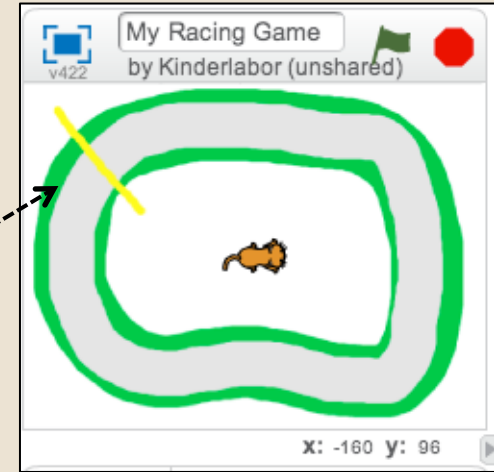
Test the ,s' key when your racing script is ready!

Scratch cat should start, run... and end up in the grass (you can't steer yet!)

Process: After the start, Scratch cat should repeatedly (i) move one step, and (ii) check whether she is touching the grass; if so, she should say „Game Over!“ and stop the race.



Choose color: First click on the colored square in the block, then on the grass! The colored square gets the color of the grass.



Use these commands:

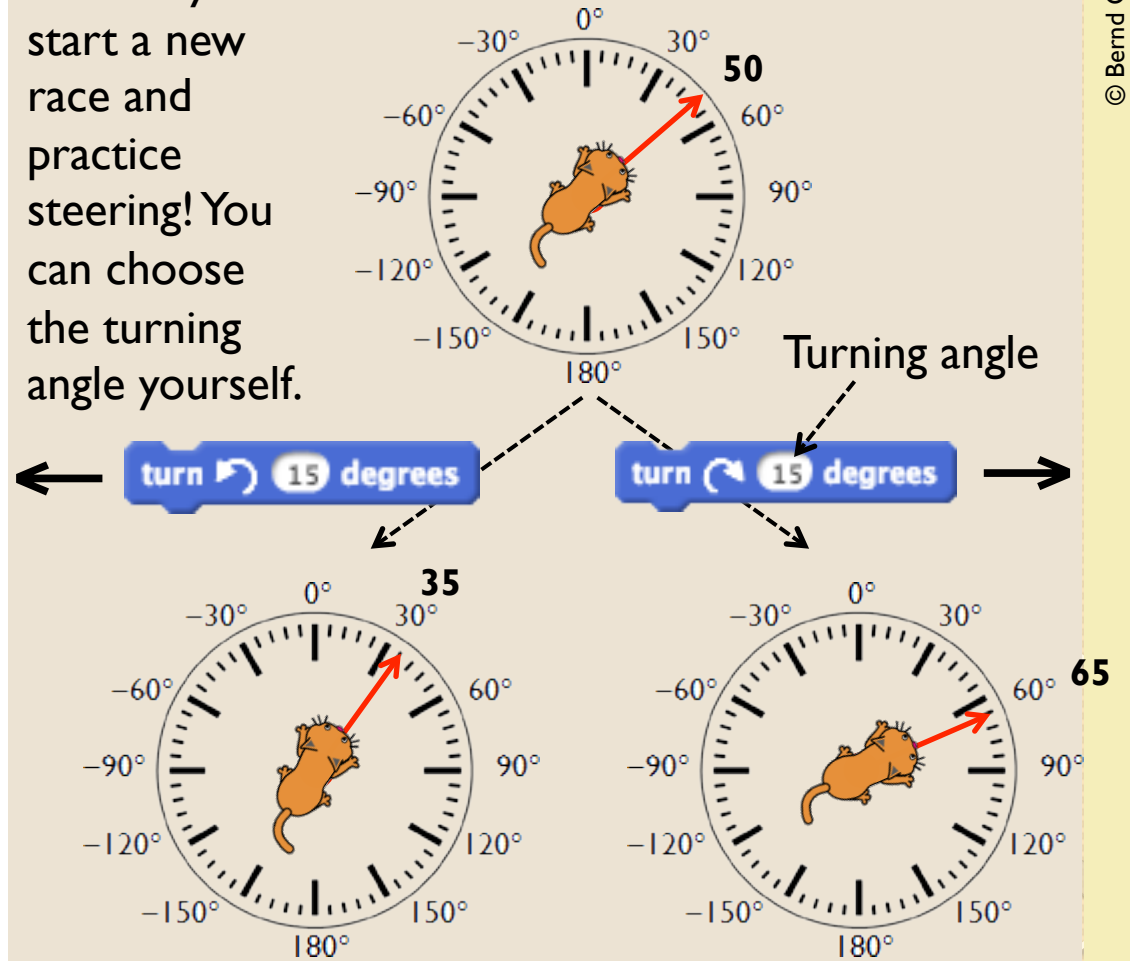


Give Scratch cat two steering scripts! The key ← should make her turn a bit to the left, and → a bit to the right!



With ,s', you can always start a new race and practice steering! You can choose the turning angle yourself.

Change direction:



Scratch cat is changing direction by -15 degrees (this is called **counterclockwise**)

Scratch cat is changing direction by +15 degrees (this is called **clockwise**)

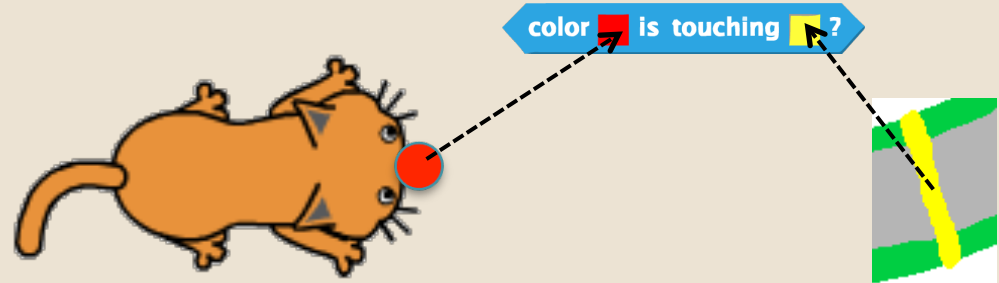
Use these commands:



Now we're almost done! Extend your racing script with a second check after every step: has Scratch cat reached the goal?



Process: Scratch cat should repeatedly check whether her nose (paint it red!) is touching the finish line! If so, she should make „meow“, say „You win!!“ and stop the race.



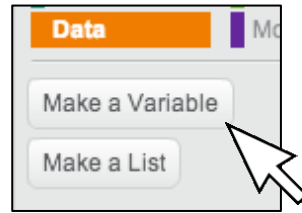
Now your game is ready! Start it with ,s' and try to steer Scratch cat to the finish line!

I still have one little puzzle for you!

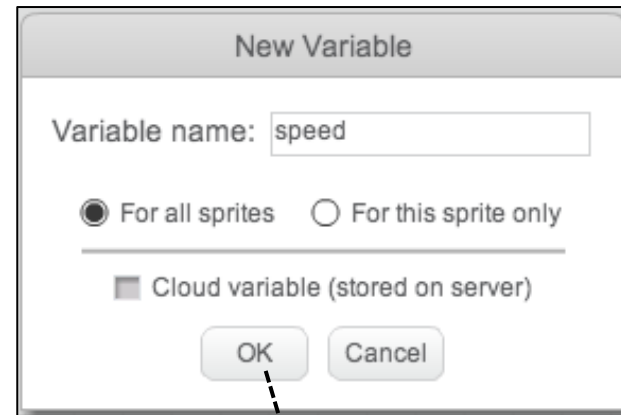
Here it is: Scratch cat is using the block ,nose at finish line?' (color is touching ?) to check for the goal. Why didn't we just give her a ,touching finish line?' (touching color ?) block for this, similar to the earlier ,touching grass' block (touching color ?)?



1. In the block palette, click **Data** and choose **Make a Variable!**



2. Select variable name (e.g. *speed*)






Congratulations, you have finished the handout „Scratch in 3 Hours“!! Here are some more hints for extending your racing game!

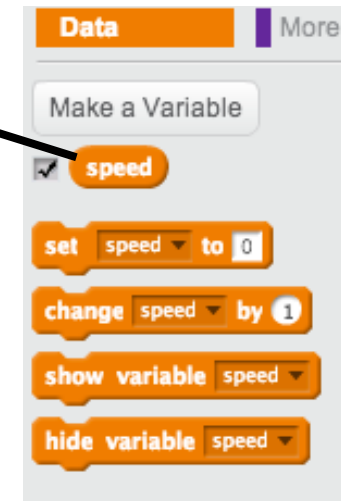
3. Instead of always moving one step, move speed steps during the race!

The new variable shows up in the block palette!



4. In the beginning of the race, set the speed to 1 ().

5. Give Scratch cat two new scripts! With key \uparrow , she should become faster (), and with key \downarrow , she should slow down ().



Time

1. Create a new variable (**time**) to measure how long the race takes!
2. In the beginning of the race, restart the timer (**reset timer**) and set the time to 0 (**set time to 0**)!
3. When Scratch cat reaches the finish line, set the time to the timer value (**set time to timer**)
4. Say the time after the race:

```
say join join You win in time seconds!! for 2 secs
```

Best time

1. Create a new variable (**best time**) to determine your best round!
2. If the best time is still zero (after the first round), or if the time of the current round is better (**best time = 0** or **time < best time**), set best time to the current time (**set best time to time**) and let Scratch cat announce a new best time!

The **lap counter** is a puzzle for experts, since it doesn't work as described below. What is the problem, and how can you fix it?

1. Create a new variable to measure the number of laps (**laps**)!
2. In the beginning, set the number of laps to zero (**set laps to 0**).
3. Each time Scratch cat is touching the finish line, increase the lap counter (**change laps by 1**) and let Scratch cat win only after a predetermined number of laps (e.g. **laps = 3**).

