

A Simple Computer Game – The Horse Race

This tutorial introduces some more sophisticated programming concepts and logic as you produce your first computer game.

Remember to save your work frequently.

- [Step 1: To the races!](#)
- [Step 2: Create a galloping horse](#)
- [Step 3: Adding randomness](#)
- [Step 4: Create a starter flag](#)
- [Step 5: Broadcasting a message](#)
- [Step 6: Hearing the message](#)
- [Step 7: Program the finish line](#)
- [Step 8: The starting position](#)
- [Step 9: Defining homebase](#)
- [Step 10: On your mark, get set](#)
- [Step 11: Go!](#)

Step 1: To the races!

In this tutorial, you create a simple horseracing game.

Start by painting a green background with a wide red finish line on the screen. The horses run when on the “grass” and stop when the first horse reaches the finish line.



Step 2: Create a galloping horse

Next, create a galloping horse. Design one horse first and, once you have added all the features, duplicate the horse to create your starting line up.

Hatch a new turtle.



Turn it to face the right (the red finish line) by dragging it:



...or by right-clicking on it, selecting **Open Backpack**, and setting its heading to 90 in the State Tab.

Animate the turtle by using one of the sets of MicroWorlds EX horse shapes. (This example uses the set of three horses.)

Click on the Painting/Clipart button:



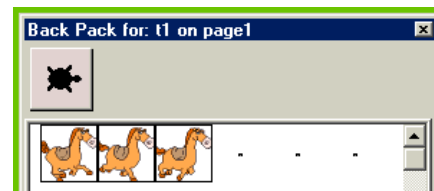
Then click on the Animation shapes button:



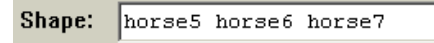
Find the three horse shapes. Click on the first shape in the set. Then hold down the Shift key and click on the last shape in the set. All the shapes are selected.

Now, drag the three shapes to the turtle. All three shapes are added to the turtle's shape list. Having a turtle use three shapes at once creates a “flipbook” type of animation.

If the turtle's backpack isn't open, tight-click on the turtle and select **Open Backpack**. Select the turtle's Shapes Tab. You should see:



Finally, go back to the State Tab in the turtle's backpack. In the shape list you should see:



Click on the Edit button next to the turtle's name and give your horse a new name - a good racing name, such as Speedy.

Step 3: Adding randomness

To animate Speedy, first click on the Rules tab in Speedy's backpack.



- In the OnClick instruction line, type:
`fd random 10 wait 1`
- Select Forever.



- Click OK.

In earlier tutorials, the input for `fd` has been a specific number (for example, `fd 5`). This time its input is another MicroWorlds EX word, `random`, which has an input of its own - 10.

`Random` takes a number as input and reports back a randomly chosen (non-negative) integer, including zero, that is less than its input. For example, `random 3` would pick 0, 1 or 2 "out of a hat" and report it to another command.

`Wait` tells MicroWorlds EX to pause for a specific amount of time. The time of the pause (`wait`'s input) is given in tenths of a second. `wait 1` causes MicroWorlds EX to pause for one tenth of a second.

Click on Speedy to animate him. He should gallop erratically because `random` picks a new number every time the instruction is run. Try different inputs for `random` until you find a nice range of motion.

Step 4: Create a starter flag

Most races have a starting signal of some kind to start a race. In this race, there's a starter flag. For this race, make a flag that waves and "yells" go when you click on it.

Hatch a new turtle. Open the turtle's backpack and click on the Shapes tab.

Next, click on the Painting/Clipart button:



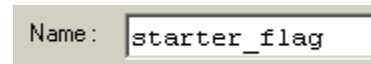
Then click on the Animation shapes button:



Shift-click on the two checkered flag shapes to select them. Drag them into the first two spaces in the new turtle's Shapes Tab. Then drag the shapes to the new turtle.

In the Command Center, type:
`setsh 1 wait 3 setsh 2 wait 3`

Give this turtle a meaningful name, for example: `starter_flag`



Remember to use one word names for turtles.

Step 5: Broadcasting a Message

The flag waves, but it doesn't "yell" anything. MicroWorlds EX does not have a yell command, but it does have a `broadcast` command.

`Broadcast` sends out a message to any turtle ready to hear it, which means any turtle with an `OnMessage` instruction.

Open the backpack of the turtle called `starter_flag` and click on the Procedures Tab. Create the following `start_the_race` procedure:

```
to start_the_race
  setsh 1 wait 3
  setsh 2 wait 3
  broadcast "go
end
```

Click on the Rules tab in the same backpack. In the OnClick instruction line, type:
`start_the_race`

Keep it set to Once.



Click on the starter flag turtle. The flag waves once and broadcasts its message, but is anyone listening?

Step 6 : Hearing the Message

To get Speedy to hear the message, he must have an OnMessage instruction.

Open Speedy's backpack. Select the Rules tab. In the OnMessage instruction line, type:
clickon



OnMessage clickon

Clickon tells the turtle to run whatever instruction appears in its OnClick instruction line, either `Once` or `Forever` (over and over again), depending on how the OnClick instruction has been set.

When Speedy "hears" the broadcast message (he doesn't care what the message is), he runs his OnClick instruction.

Step 7: Program the Finish Line

What's wrong with the race? Speedy never stops! Program Speedy to stop when he touches the color red – the color of the finish line – so that the race ends when Speedy crosses the finish line.

- o Right-click on Speedy and open his backpack
- o Open the Rules Tab
- o Select the color red in the OnColor drop down color menu.
- o In the instruction line, type `winner`. Leave the OnColor instruction set to `Once`.

You just introduced a new word, `winner`. Open Speedy's Procedure tab and enter the `winner` procedure:
to winner
 everyone [clickoff]
 announce [I win!]
end

Everyone makes all the turtles on the current page run the instruction that is in the square brackets, one after the other. (Although there are no other horses in your race now, there soon will be.)

In this case, when a horse touches the finish line, everyone stops moving and the winning horse announces that it wins.

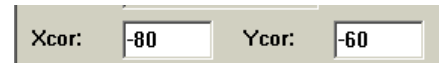
Click on the starter flag turtle to test Speedy.

If you're satisfied with Speedy, use copy and paste to create three more horses for the race. Give all your turtles good "racing horse" names. For names with more than one word, push the words together or use underscores or periods between them (BigShot, Big_Shot, or Big.Shot).

Step 8: The starting position

Drag Speedy to the left edge of the screen to a place you would consider a good "starting line" position. Be sure Speedy is entirely on the page.

Right-click on Speedy and select **Open Backpack**. Click on the State tab. Note Speedy's `xcor` (x co-ordinate). The settings may look like this:



Xcor: -80 Ycor: -60

Every position on the page has both an x co-ordinate and a y co-ordinate. The center point of the page is position [0 0].

Move all the other turtles to their starting line positions. Open each turtle's backpack and check that the `xcor` for each turtle is the same.

You may want to space the horses evenly along the y-axis by setting each turtle's `ycor` (y co-ordinate) appropriately. You may also want to check that each turtle's heading is set at 90 so that it will run in a straight line towards the finish line.

You will use these positions to put the horses at the starting line before each race.

Step 9: Defining homebase

In the Command center, type:
turtlesown "homebase

The `turtlesown` command gives the turtle a new property or state. All turtle's already have a heading, an `xcor`, a `ycor`, a pen state (pen up, pen down, or pen erase), color, and shape. Now the turtles in your project also have a `homebase` property.

Whenever you add a `turtlesown` property, two "dynamic primitives" are created. These are primitives that exist only in the project in which the new `turtlesown` property has been added and they only exist as long as this new property exists.

The first dynamic primitive is a command that sets the value of the property. It consists of `set-` and the name of the property, in this case, `sethomebase`.

The second dynamic primitive is the name of the property. It's a reporter – it reports the value of the property. In this case, the reporter is the word `homebase`.

Look at the bottom area of Speedy's State Tab.



The turtle has a `homebase` property, but no "value" is linked to this property. To set the value, type the following instruction:

```
Speedy, sethomebase [-180 30]
```

(Use the values for `xcor` and `ycor` that are in your horse's State Tab. Remember to use your horse's name.)

`Speedy, means talk only to Speedy`. To talk to only one turtle, use that turtle's name **and** a comma after it. (You must use the comma.)

`Sethomebase` sets the value of the `homebase` property. It is set to the co-ordinates listed in the square brackets.

Set a `homebase` for each horse. Remember to use each horse's name (followed by a comma) and the `xcor` and `ycor` in the horse's backpack. Set a `homebase` for the `starter_flag` turtle also, in case it's moved.

```
starter_flag, sethomebase [150 185]
```

Step 10: On your mark, get set

Write a procedure to set up the race and start the action.

Since this is a procedure that is used to run the action in the whole project, write the procedure in the project Procedures Tab (*not* in the Procedures Tab of a backpack).

The procedure should tell all the turtles to go to their starting position and `starter_flag` to `clickon`.

```
to setup
  everyone [setpos homebase]
  starter_flag, clickon
end
```

`Setpos` tells MicroWorlds to set the turtle's position to whatever the input is. The input must

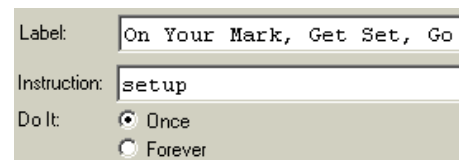
be a set of co-ordinates (such as `[0 0]`) or a reporter that reports a set of co-ordinates. In this case, the input is each turtle's `homebase`.

Add a button to make set up easier.

Click on the button tool in the Tool bar:



Click anywhere on the page. A dialog box opens. Give the button a meaningful label. In the field next to Instruction, type `setup` (use the name of your set up procedure). Leave Do It set to `Once`.



Click OK. Make your button larger, if necessary, by selecting it (either drag around it or press CTRL and click on it) and then dragging one of the corners.

Place your button anywhere on the screen.

Step 11: Go!

Now, start your race.

Can you predict which horse will win?

You may want to try some of these ideas :

- Change the shape of the winning horse.
- Think of a way to give the horses different speeds.
- Create other sorts of races.
- Keep track of the number of times each horse wins. Draw a graph showing this information. Change the speed of one horse to see how this affects your results.
- Use the Melody Editor to add music to the race or Sound Recording to add your voice.