Step-by-Step Epidemic Model Tutorial

This tutorial has step-by-step instructions to help you build an epidemic model using ToolBlox. This tutorial is appropriate for someone with no programming experience. However, if you do not know your way around the ToolBlox interface, it might be good to do the ToolBlox Orientation Activity (http://54.221.193.36/media/cms_page_media/8/1.%20Orientation.docx) first.
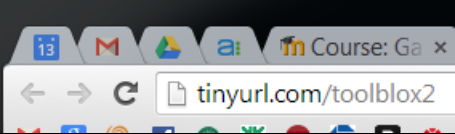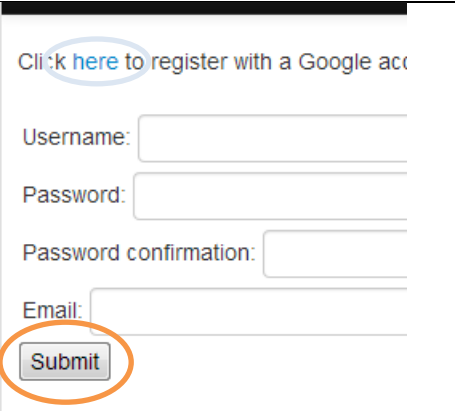
In this model, hundreds of agents all randomly wander around the world. "Infected" (red) agents spread disease when they collide with "uninfected" (yellow) agents. Over time, infected agents get better and become "immune" (blue) which means they cannot be re-infected.
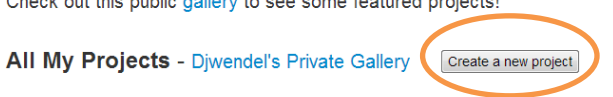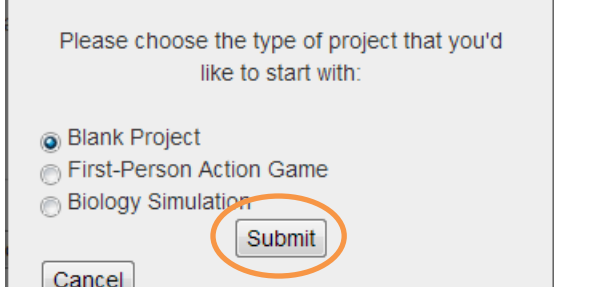
Contents

1. Create a ToolBlox Account

If you do not already have a ToolBlox account, you will need to create one now.

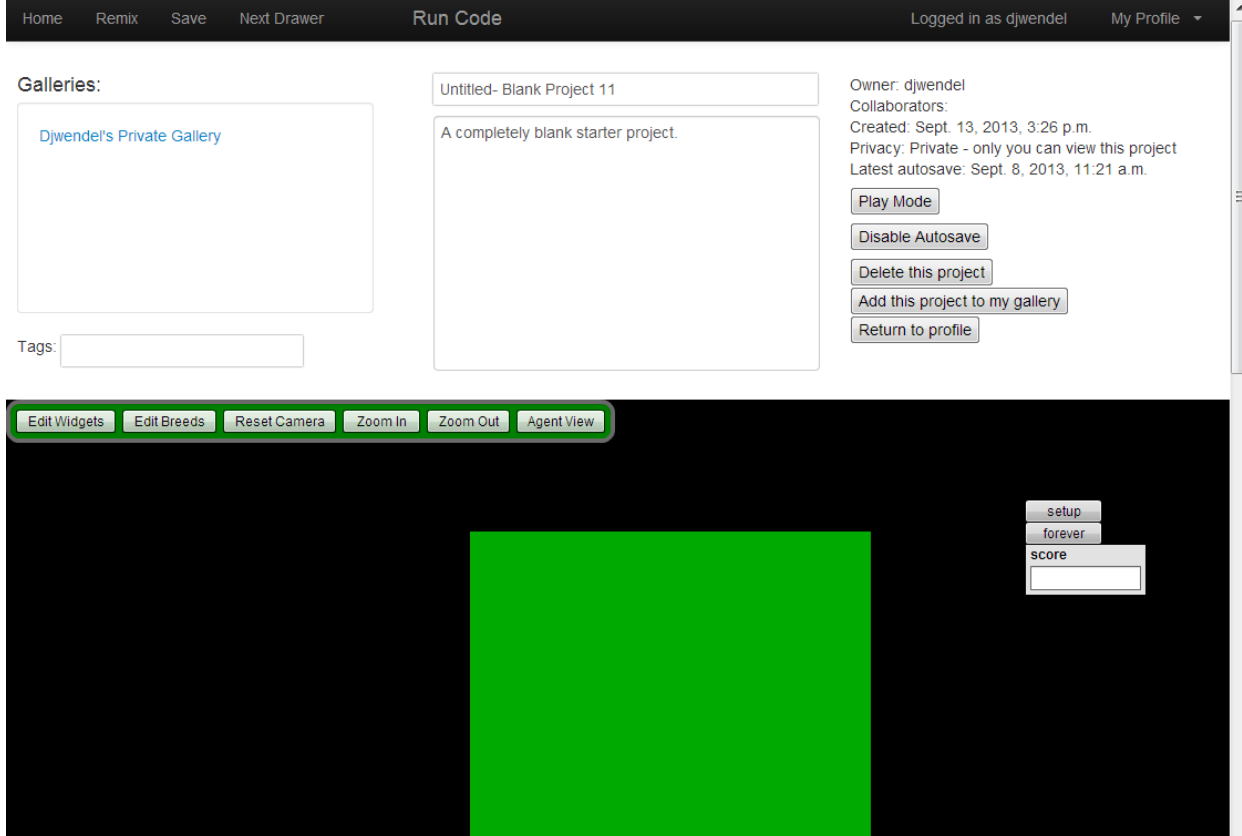|	
tinyurl.com/toolblox2	ToolBlox is located at http://tinyurl.com/toolblox2 which will redirect you to the ToolBlox page.
Welcome to ToolBlox! Please log in or register for an account Updates:	
Click here to register with a Google account Username: Password: Password confirmation: Email: Submit	After filling in the username and password you want, click the Submit button to create your account. OR If you want to use a Google account to log in, click the blue link at the top of the page.

2. Create a New Project

Once you have successfully created an account or logged in, follow these steps to create a new blank project.

Logged in as djwendel My Profile ▾	If you are not already on your profile page, click My Profile to go there.
Check out this public gallery to see some featured projects! **All My Projects** - Djwendel's Private Gallery Create a new project	
Please choose the type of project that you'd like to start with: ● Blank Project ○ First-Person Action Game ○ Biology Simulation Submit Cancel	We will be starting from a blank project, so choose that option.

Check your work

You should now see a new blank project in the ToolBlox programming editor:

| Home | Remix | Save | Next Drawer | Run Code | Logged in as djwendel | My Profile ▾ |

Galleries:

Djwendel's Private Gallery

Tags:

Untitled- Blank Project 11
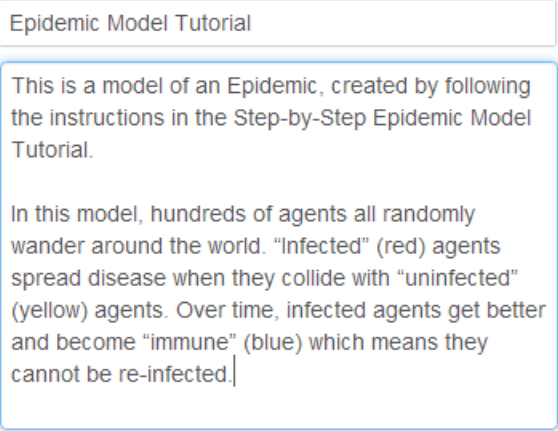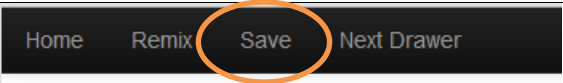
A completely blank starter project.

Owner: djwendel
Collaborators:
Created: Sept. 13, 2013, 3:26 p.m.
Privacy: Private - only you can view this project
Latest autosave: Sept. 8, 2013, 11:21 a.m.

Play Mode

Disable Autosave

Delete this project
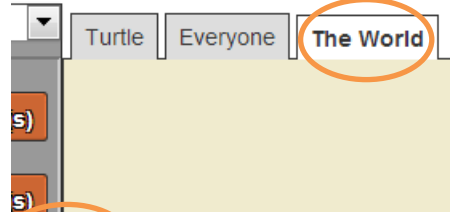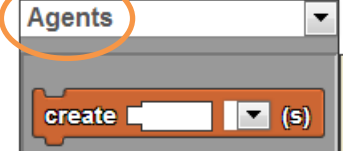
Add this project to my gallery

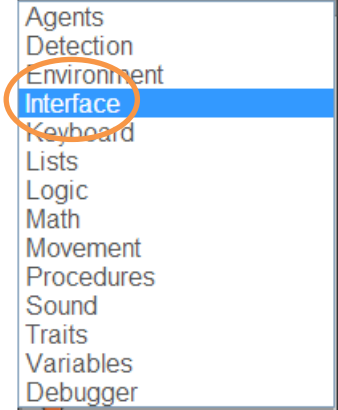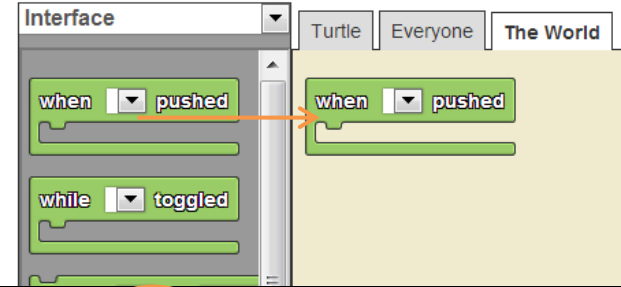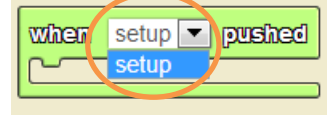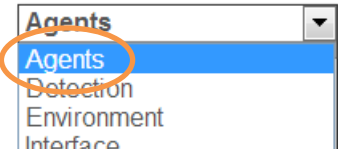Return to profile

Edit Widgets   Edit Breeds   Reset Camera   Zoom In   Zoom Out   Agent View

setup
forever
score

3. Rename Project and Save
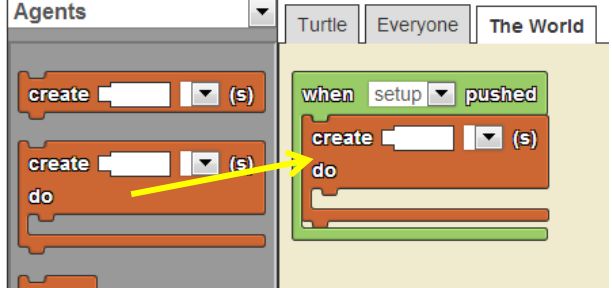
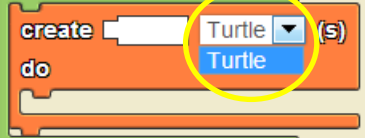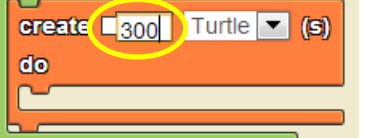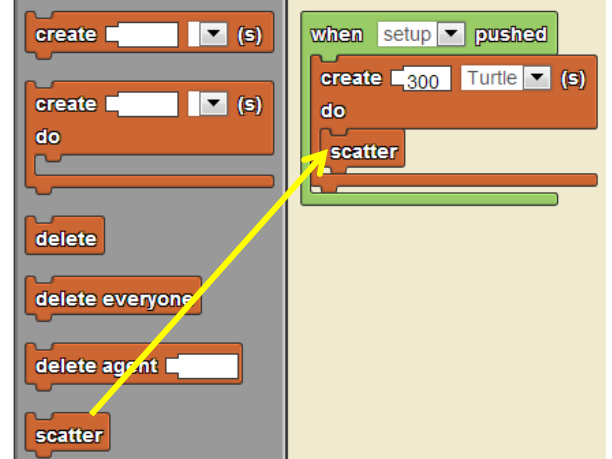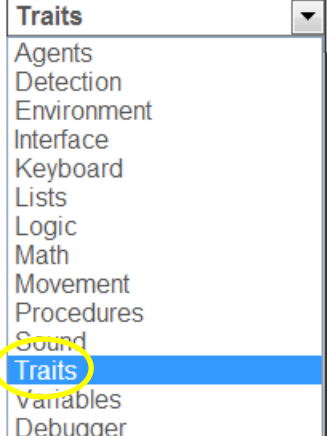The default name and description for your project are not very descriptive. Change them to something more useful.

Epidemic Model Tutorial	
Epidemic Model Tutorial This is a model of an Epidemic, created by following the instructions in the Step-by-Step Epidemic Model Tutorial. In this model, hundreds of agents all randomly wander around the world. "Infected" (red) agents spread disease when they collide with "uninfected" (yellow) agents. Over time, infected agents get better and become "immune" (blue) which means they cannot be re-infected.	
Home Remix Save Next Drawer	Clicking save causes your project to "manual save." The project also auto-saves every 30 seconds.

4. Create 300 Agents

The epidemic model takes place in a population of Turtle agents. To make this population, create and scatter 300 Turtles, and set their color to yellow.

	Switch to The World page. Because only The World exists at first, it is the one who needs to create all other agents.
	Click on the drawer title to show a drop-down box to select a different drawer.
	We want the Interface drawer.
	The **when___pushed** block will tell the World to do whatever commands we attach inside it when the user clicks the button that we choose from the drop-down.
	We choose "setup" from the dropdown. So The World will do whatever commands we attach inside this block when "setup" is clicked.
	Now switch to the Agents drawer to find blocks for creating agents.

	Choose a **create-do** block, because we want to create some agents and then tell them to immediately change something (their color).
	Select Turtle from the dropdown.
	And tell it how many Turtles to create.
	Scatter tells the newly-created agent to move somewhere random.
	The blocks for changing agents' colors are in the Traits drawer.

	Attach the **set my ___** block inside the **create-do** block, after the **scatter** block. Attach a **color** block into the **set my ___** block.
	Choose color from the drop-down because we want to set the agent's color.
	Choose yellow from the color drop-down; yellow represents and uninfected agent.
	Click Save to save your work so far.
	You need to click Run Code to turn the blocks you connected into machine code that the agents can run.

Check your work
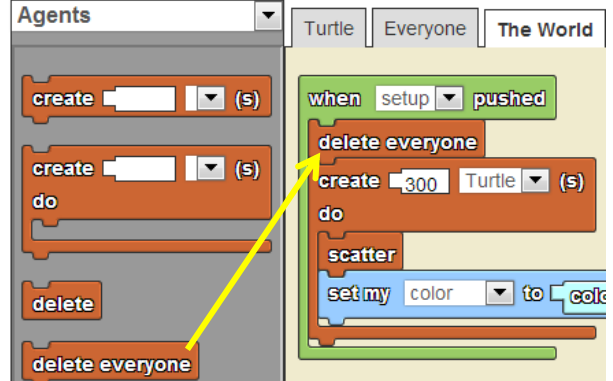
Click the Zoom In button twice to make the 3D world a little bigger. Then click setup . The World should follow the commands you just programmed, and create 300 yellow Turtles in the world. The Turtles will look like cubes. It should look something like this:
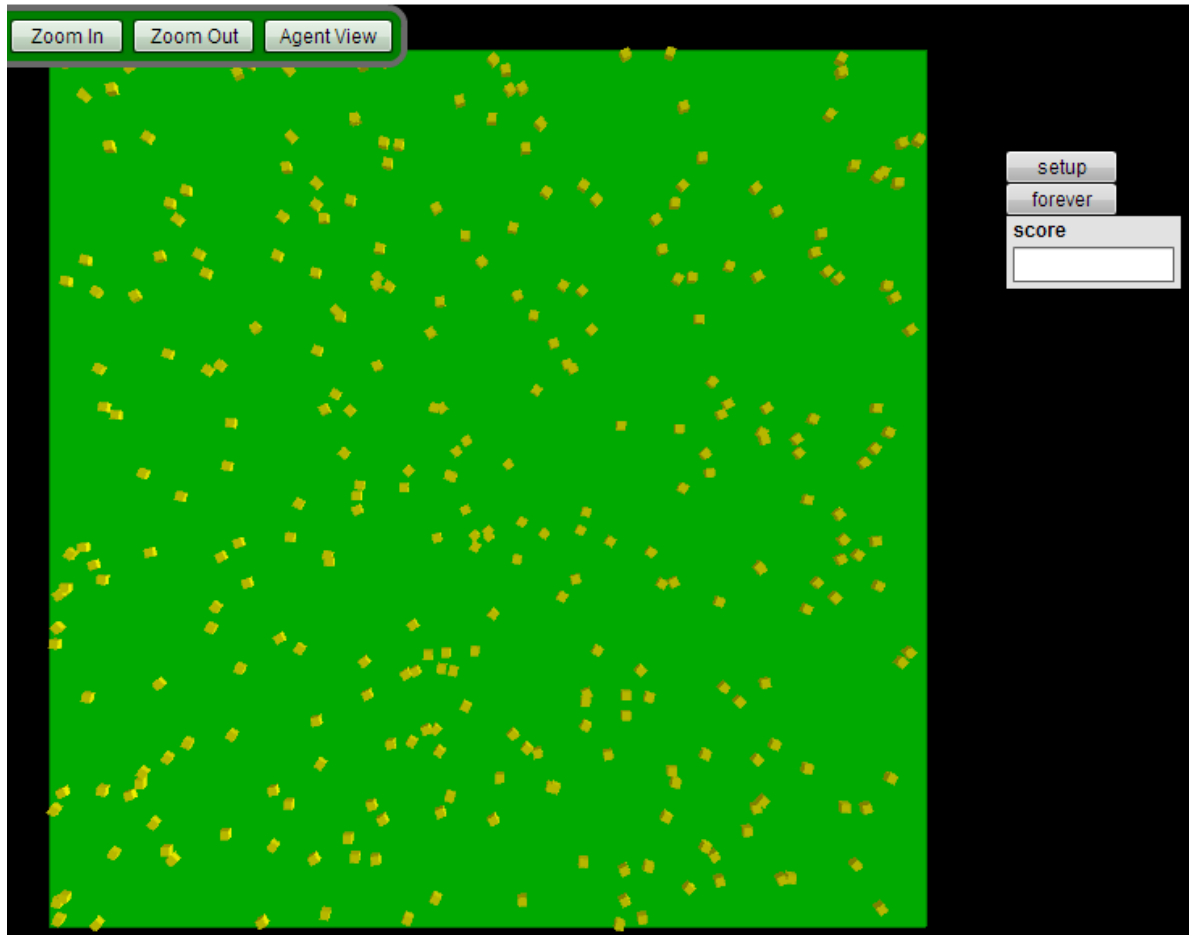
5. Clear Old Agents Before Creating New Ones

If you click [setup] again you will notice that now there are 600 Turtles in the world rather than just 300. Oops! To fix this, we need to add a block to clear out the old agents before creating the new ones.

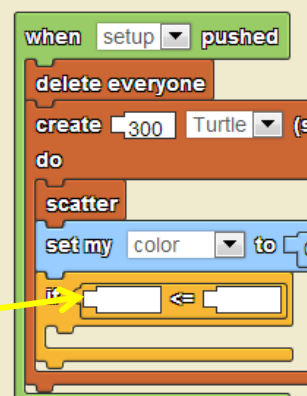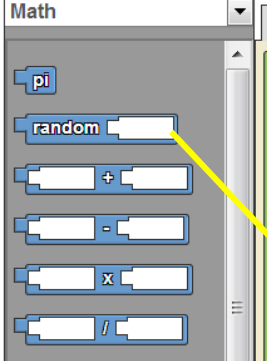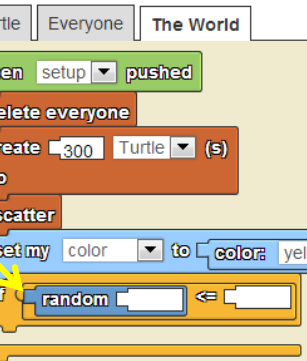|	
	Attach a **delete everyone** block BEFORE the **create-do** block. Otherwise, it will delete all of the agents that you just created! Most programs have setup code like that this starts by deleting everyone and then creating some agents.
	Don't forget to save.
	Because you changed your program, you need to click Run Code again.
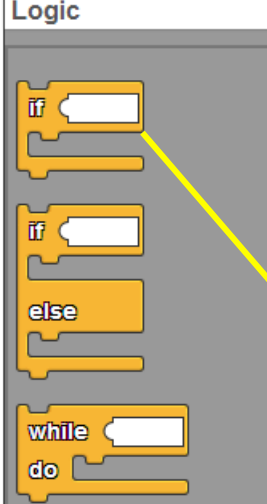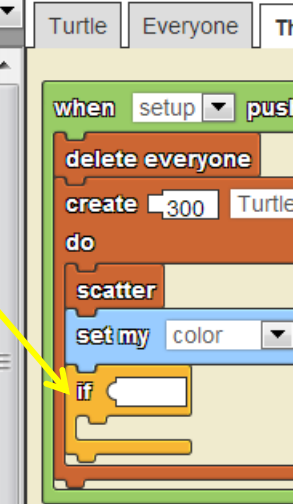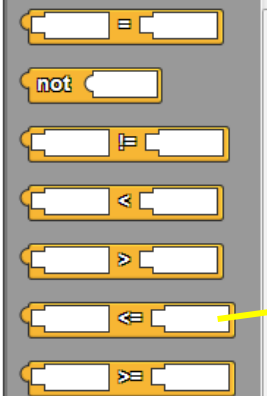
Check your work

Now if you click [setup] you should only see 300 Turtle agents, no matter how many times you click.
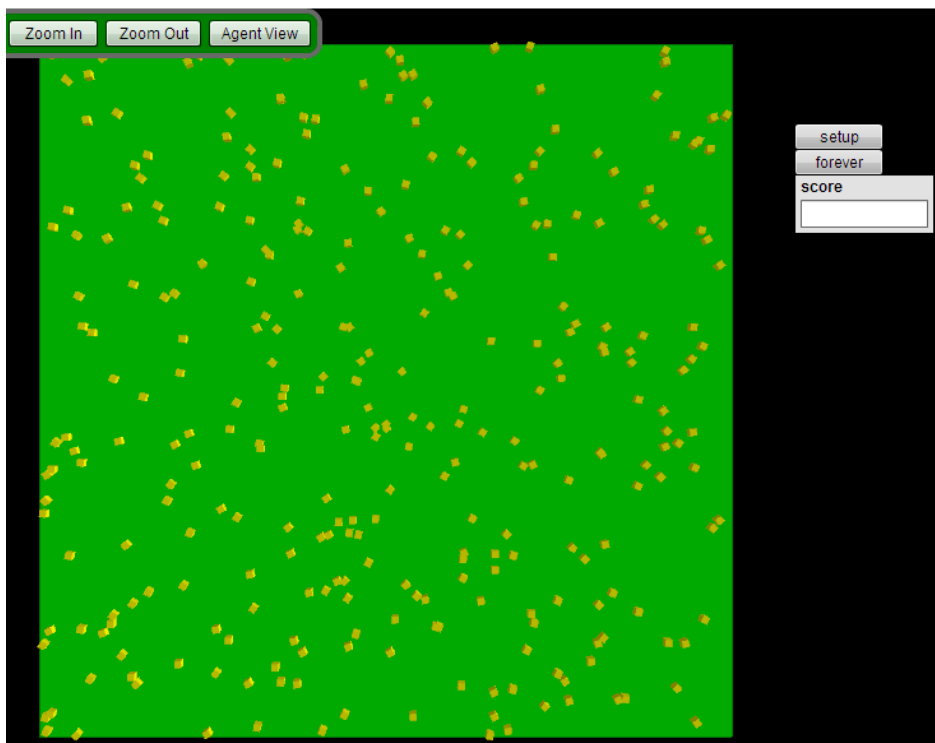
6. Infection

In this model we will make 10% of the Turtle agents start out infected. We will represent "infected" by setting the Turtles' color to red.

	The **random** block gives a number between 0 and the number you give it - 1.
	Random 100 will be less than or equal to 10 10% of the time. This will cause the commands inside the **if** block to run about 10% of the time.

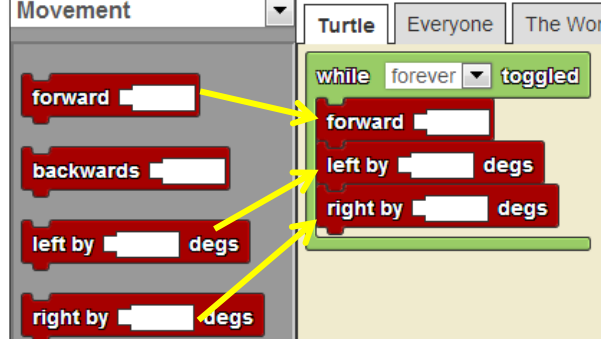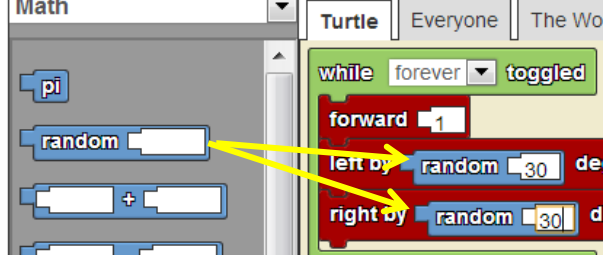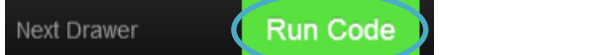|	
	In the Traits drawer, find the blocks to set the agent's color to red.
	Don't forget to save…
	…and Run Code after every change you make.
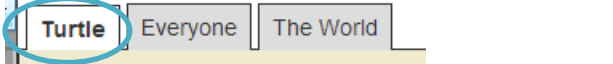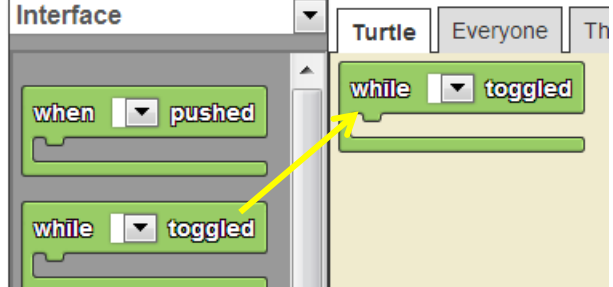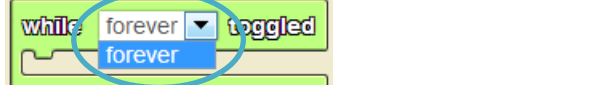
Check your work

Now if you click you should see 300 Turtle agents, with about 10% of them being red and the rest being yellow.
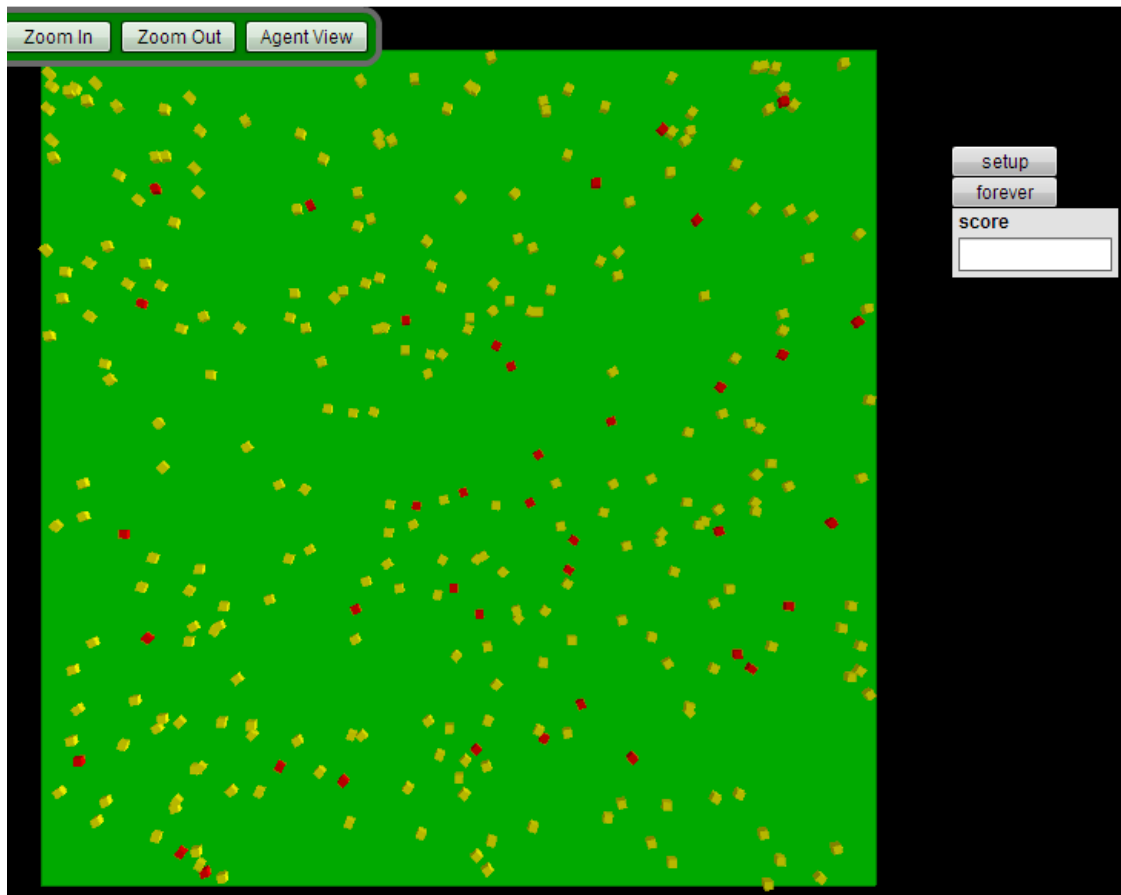
7. Movement

Now it is time to program the behavior of the Turtle agents as the simulation runs. First we will make them wander randomly while the [forever] button is selected.
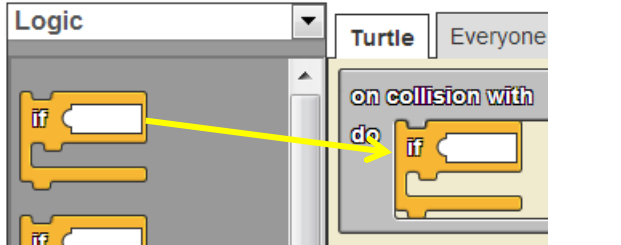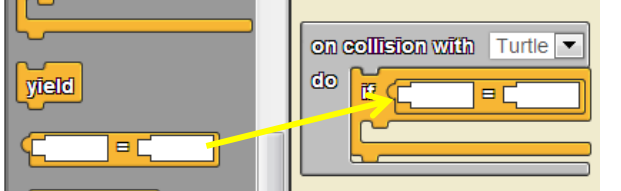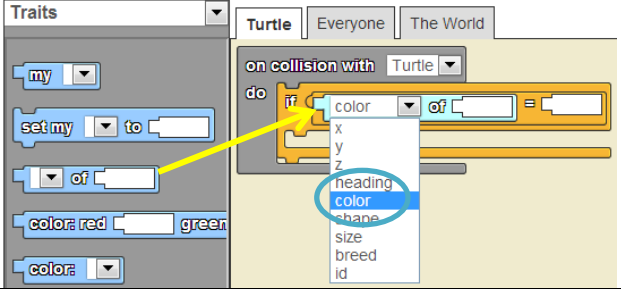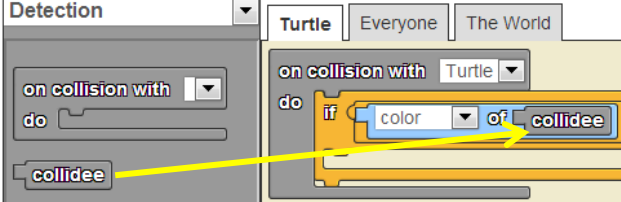
	Because we want the Turtles to do the new commands, we switch to the Turtle page.
	The **while___toggled** block will run whatever blocks are inside it, as long as the button you choose from the drop-down is currently "toggled" on.
	Choose "forever" from the drop-down to make this block respond to the [forever] button.
	To make the Turtles move randomly, first get some movement blocks from the Movement drawer.
	Tell the Turtles to move forward 1 step at a time, and to turn both left and right by random 30. This will make them sometimes go slightly left, and sometimes slightly right.
Home Remix Save Next Drawer	
Next Drawer Run Code	

Check your work

Now if you click [forever] you should see the Turtles moving in random paths, mostly straight but with a little random variation side to side.

8. Contagion via Collisions

Currently the infection does not spread through the population. To make it spread, program the Turtles to turn themselves red whenever they collide with another Turtle who is already red.

	To make turtles do something when they collide with another turtle, first drag a collision block onto the Turtle page.
	Then select Turtle from the drop-down. Now, when a Turtle collides with another Turtle, it will do the commands you attach here.
	We need an **if** block because we want the turtle to turn red IF the one it collides with is also red.
	We will use = to see if the other Turtle's color = the color red.
	We want to check to the **color of**…
	… the other Turtle, which is called the **collidee**.

	Use the **color** block when specifying colors. Just typing the name of a color into the **=** block won't work.
	Now find the blocks to set the Turtle's color to red. We attach this inside the **if** block, so they will only run if the test of the **if** block is true.

Check your work

Click forever . Now when a green Turtle collides with a red one, you should see the green one turn red.
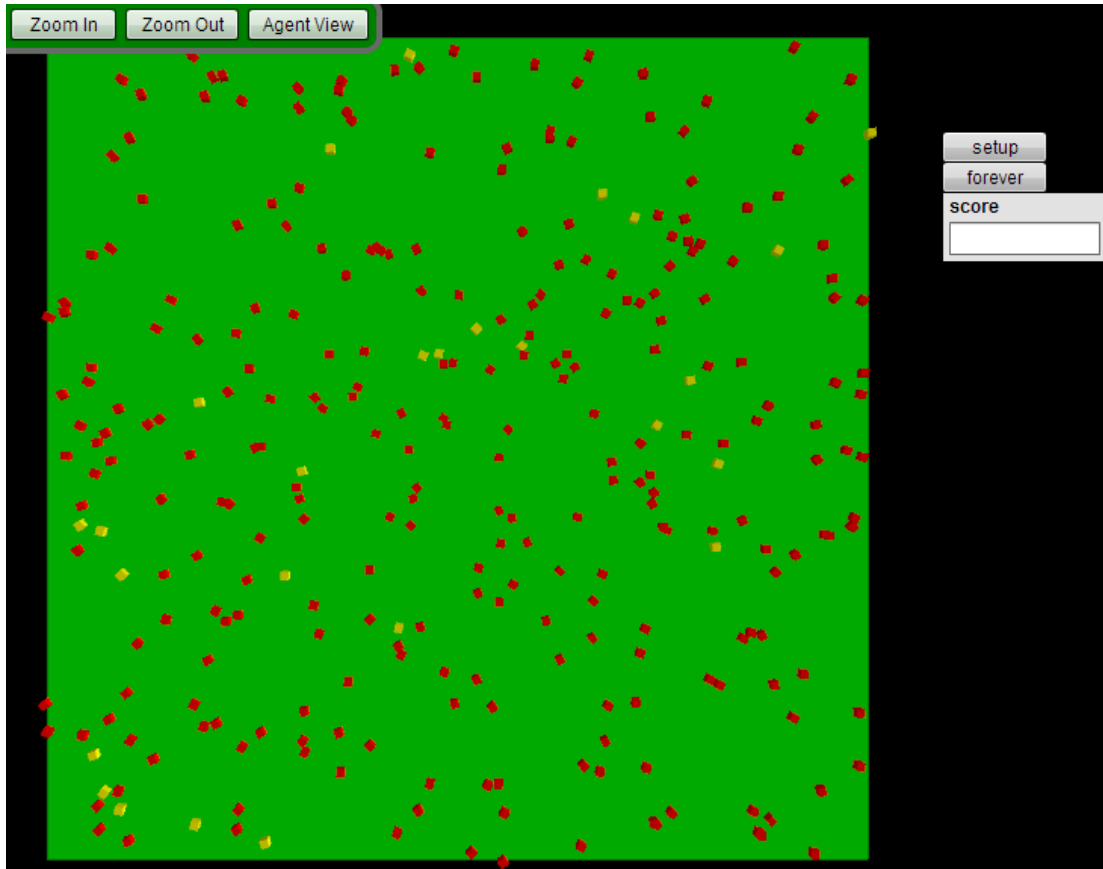
9. Recovery

In order to simulate recovery, we will give the Turtles as random chance to turning from red ("infected") to blue ("recovered").
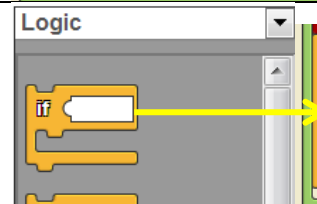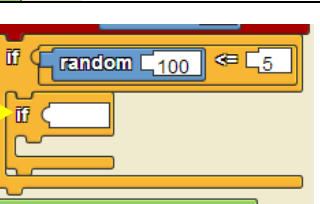
	We'll use an **if** block (with **<=** and **random**) to make the recovery code only run with a small probability.
	Random 100 will be <= 5 about 5% of the time.
	Now we need *another* if block, because we only want sick agents to recover, so we have to first check to make sure this Turtle is sick.

	A turtle is sick if its color = red, so we will be using those blocks.
	To "recover," the Turtle sets its color to blue to indicate that it has recovered and is no longer infected.

Check your work

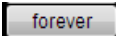Click . You should see red turtles turning blue randomly.
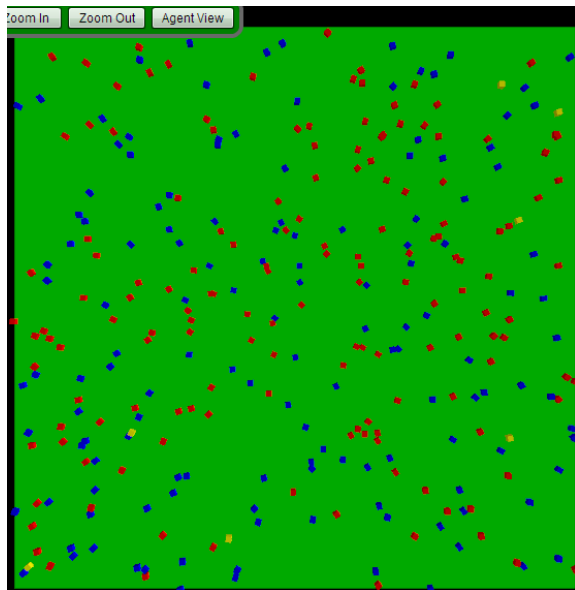
10.　Immunity

Notice that currently if a blue turtle collides with a red turtle, the blue one will turn red, meaning that it has been re-infected. In order to simulate immunity in the recovered turtles, we will modify the collision code so that only yellow turtles can be infected by red turtles. This way, blue turtles will be immune to infection.

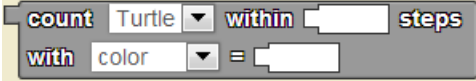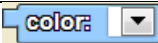|	
	We'll use another **if** block to do a second check so that the Turtle will turn red if the color of the collidee is red *and* the Turtle's color is yellow. If its color is blue, nothing will happen.
	In order for this second check to work, move the code for setting the Turtle's color *inside* the new **if** block.
	Now attach an **=** block.
	And fill it in so that **my color** has to be **=** to the **color** yellow for this **if** block to run the code inside it.

Check your work

Click **forever** . Now, blue turtles should stay blue even when they collide with red turtles.

11.   Extensions

Here are a few suggested extensions you can try to add to the model. Below each extension description is a table of a few blocks that might be helpful in completing the extension.
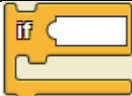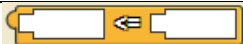
Track Immune Agents

Use the score data box to display the current number of immune (blue) agents.

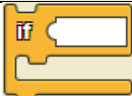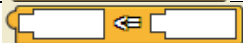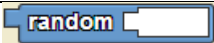| Drawer	Block	Description
Interface	set [▼] data box to [ ]	Set the contents of the selected data box widget to whatever is given.
Detection	count Turtle [▼] within [ ] steps with color [▼] = [ ]	Count the number of agents of the selected breed, within the given number of steps, whose selected trait matches the given value.
Traits	color: [▼]	The color that is selected in the drop-down.

Probability of Contagion

Modify the collision code so that Turtles only sometimes catch the infection if they collide with an infected other Turtle.

Drawer	Block	Description
Logic	if [ ]	If the given condition is true, do the commands attached inside.
Detection	[ ] <= [ ]	True if the first given number is less than or equal to the second given number.
Math	random [ ]	A random number between 0 and the given number - 1.

Death from Infection

Give Turtles a random chance of "dying" (deleting themselves) if they are infected.

Drawer	Block	Description
Logic	if [ ]	If the given condition is true, do the commands attached inside.
Detection	[ ] = [ ]	True if the two things given are equal.
Detection	[ ] <= [ ]	True if the first given number is less than or equal to the second given number.
Math	random [ ]	A random number between 0 and the given number - 1.
Agents	delete	Delete myself (make myself disappear).
Traits	color: [▼]	The color that is selected in the drop-down.