



Scratch Programming

Lesson 2-2

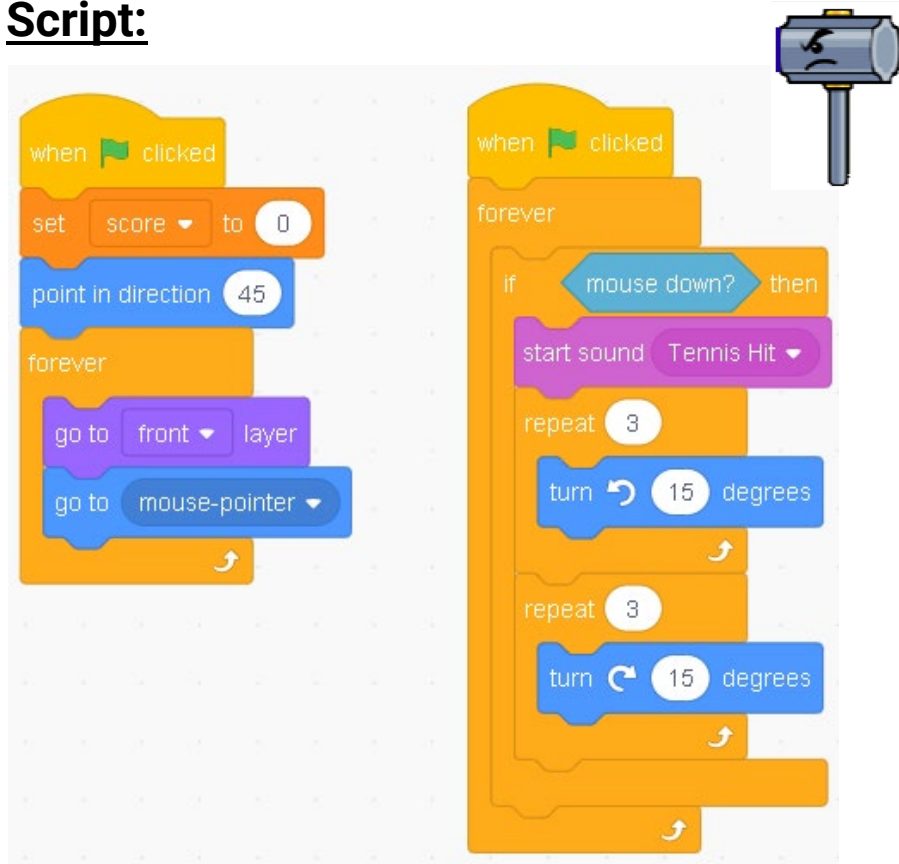
Whack-a-mole Game II

Presented by Advaspire Team



Review – Program the Hammer

Script:



Let's start with the score of 0.

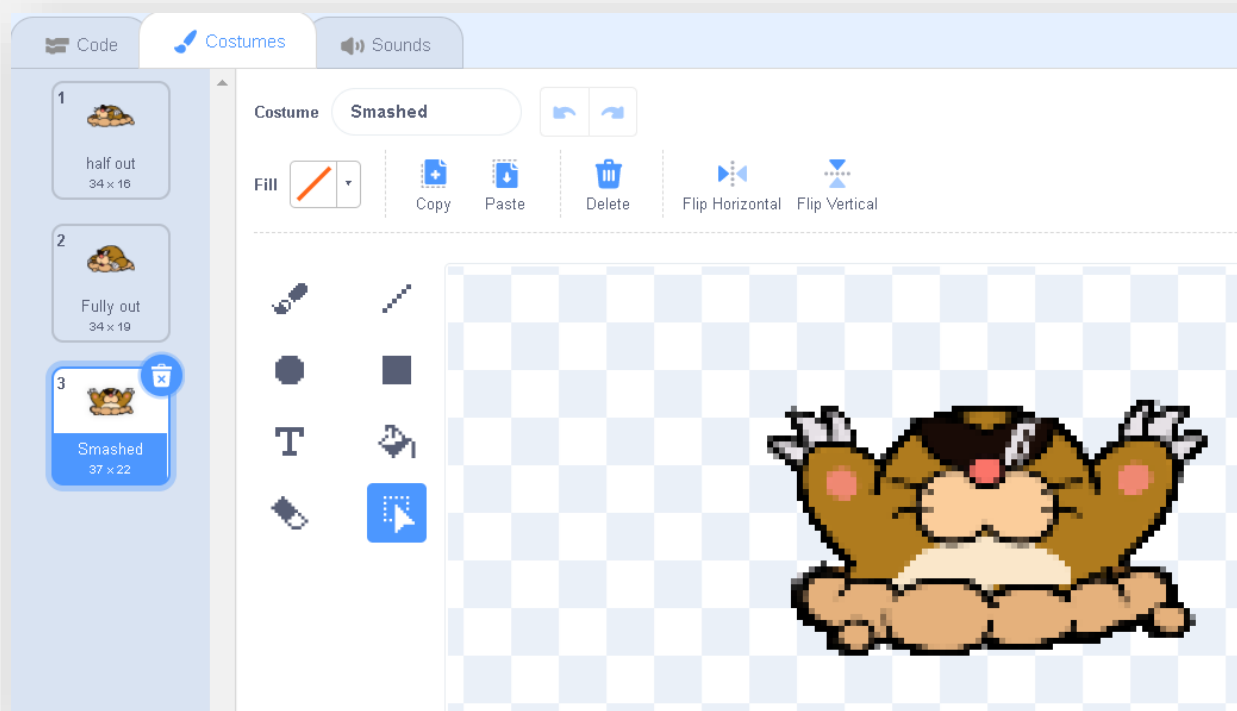
Then we will have some functions for this hammer.

1. The hammer will always go to our mouse-pointer, and must always be in front of the moles and holes.
2. When we click on the screen, it will smash, we will use turn to make the smash action.
3. Remember to add a sound effect for the smash action.

Once done, you can press start to try your hammer and see if it's working.



Review – Cut out the mole from the sprite

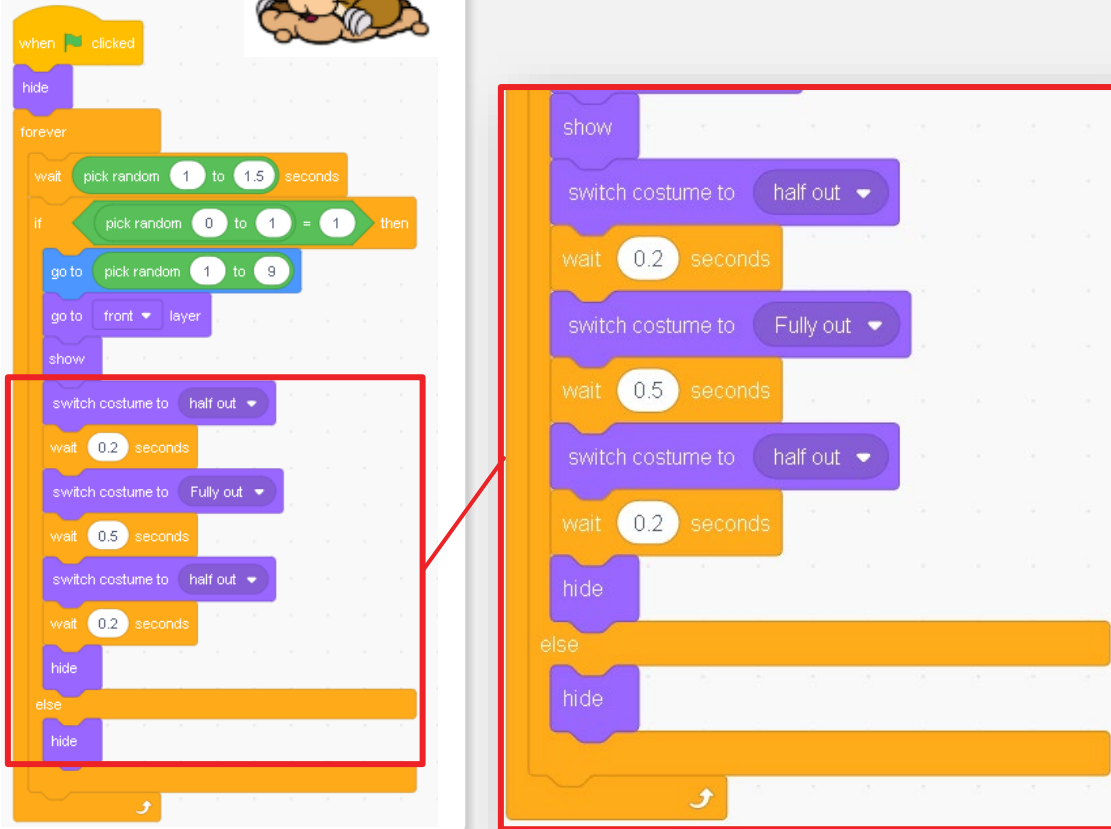


For first 2 costumes, just do like what we did to the hole, but for the costume 3 (smashed), we need to cut-off half body and add in the holes to it.



Review – Program the mole

Script:



Then we make an animation to make the mole to come out from the hole, then wait for 0.5 seconds, then move back to the hole.

And the mole has 50% of possibility to come out, it has another 50% remain hidden underground.



Review – Program the mole (Smashed)

Script:

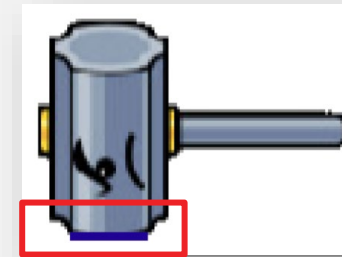
The image shows two parallel Scratch scripts for a mole character. The mole is depicted as a brown mole with a pink nose, wearing a small hat, and is currently in a 'Smashed' state (costume 3). The scripts are as follows:

Script 1 (Left):

- when green flag clicked
- hide
- forever loop:
 - wait pick random 1 to 1.5 seconds
 - if pick random 0 to 1 = 1 then:
 - go to pick random 1 to 9
 - go to front layer
 - show
 - switch costume to half out
 - wait 0.2 seconds
 - switch costume to Fully out
 - wait 0.5 seconds
 - switch costume to half out
 - wait 0.2 seconds
 - hide
 - else:
 - hide

Script 2 (Right):

- when green flag clicked
- forever loop:
 - if mouse down? and touching color dark blue?
 - start sound Boing
 - switch costume to Smashed
 - change score by 1
 - wait 0.5 seconds
 - switch costume to half out
 - wait 0.2 seconds
 - hide



I have made the front of the hammer dark blue color. Only if the player uses the hammer to smash on the mole, it will score.

Once the mole gets smashed, it will change to 3rd costume (smashed), and go back into the hole. I add in the sound effect when it gets hit.



Review – Add timer & Score to Hammer

Script:



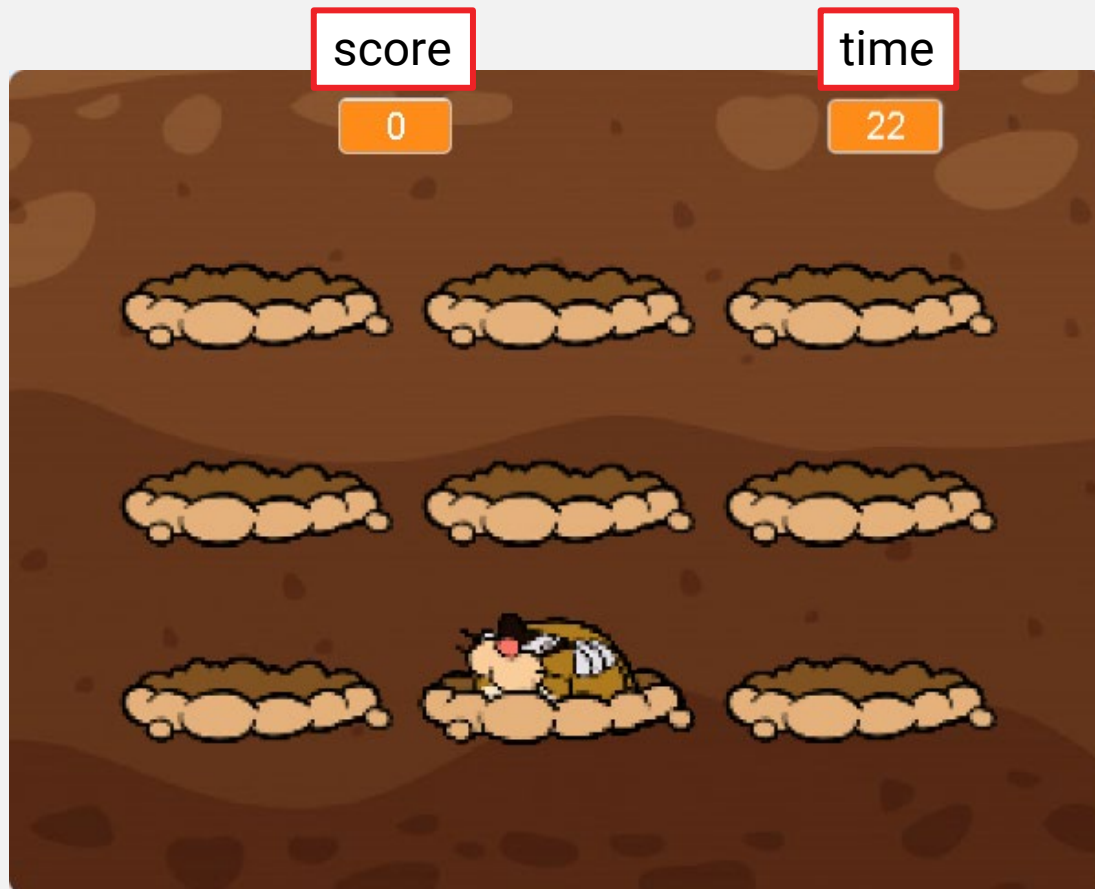
Let's set the game play time to 30 seconds.

And the game rules are:

1. Your game play only last for 30 seconds.
2. Once 30 seconds have passed, it will check the score to determine if you win or lose.
3. If you score more than 5 (each success smash will earn you 1 point), you will win the game, you lose otherwise.



Review – Set up the Game view



Then arrange the score and time variable like this (as left picture).

Score at the left, time at the right.

Solution – L2-1



Create a whack-a-mole game which has 9 holes in total.

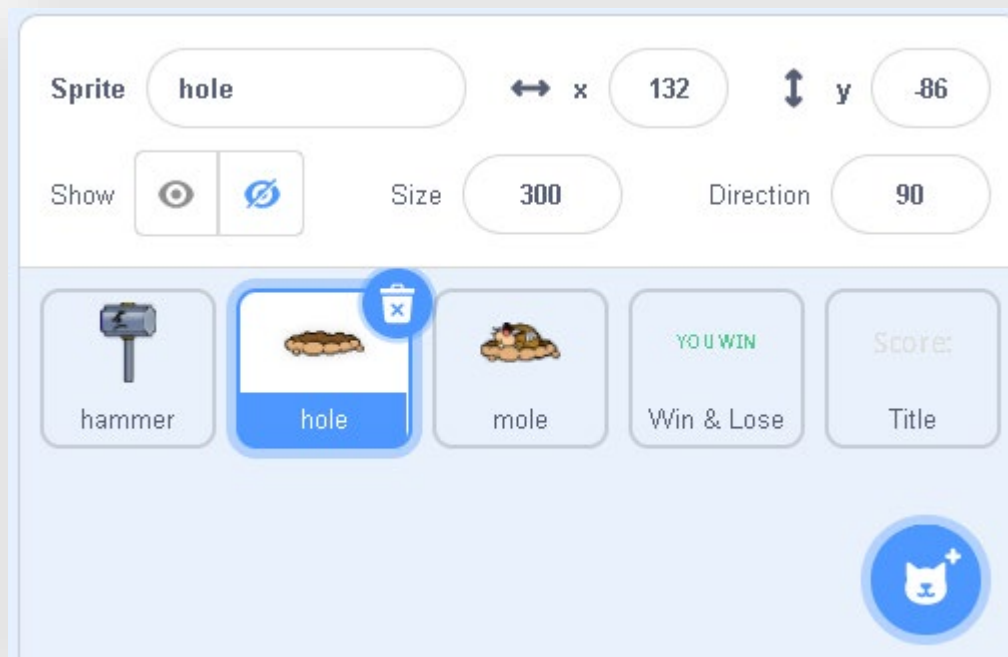
The player needs to use the hammer to hit the mole which comes out from the hole to score 1 point.

Player needs to get at least 9 points within 30 seconds of play time.

Please use create clones for the holes.



Solution – Create clones for holes



Now our challenge is to make those holes into 1 sprite only, then use create clones to set up 9 holes in the game.



Solution – Create clones for holes

Script:

The script consists of two event listeners:

- when clicked:**
 - hide
 - go to x: -132 y: 86
 - create clone of myself
 - go to x: 0 y: 86
 - create clone of myself
 - go to x: 132 y: 86
 - create clone of myself
 - go to x: -132 y: 0
 - create clone of myself
 - go to x: 0 y: 0
 - create clone of myself
 - go to x: 132 y: 0
 - create clone of myself
 - go to x: -132 y: -86
 - create clone of myself
 - go to x: 0 y: -86
 - create clone of myself
 - go to x: 132 y: -86
 - create clone of myself
- when I start as a clone:**
 - show

You need to create total of 9 holes in your game.

You can copy the positions from other holes and put them all into 1 hole, then use create clones to create them.

When it starts as a clone, it will show itself.



How do we set up for our mole?

Compare to last lesson, this time I will make “Spawn a Mole” broadcast to spawn the mole.

So at start I will call out the broadcast. Then when the mole gets hit by my hammer, it will start “Boing” sound effect then changed costume to “hit”. Then I will stop the “Spawn a Mole” script first, so that the changing costume in “Spawn a Mole” script and the forever-loop (hit by hammer) won’t clash.

It will cut off the sequence in “Spawn a Mole” and run the code to make the mole back into the hole.

Script:

The image shows two Scratch scripts for a mole character. The first script, titled "Script:", is triggered by "when I receive 'Spawn a Mole'". It contains a "wait 0.5 seconds" block, followed by a "forever" loop. Inside the loop, there is a "wait pick random 1 to 1.5 seconds" block, followed by an "if pick random 0 to 1 = 1 then" block. The "if" block contains a sequence of actions: "go to hole", "go to front layer", "show", "switch costume to coming out", "wait 0.2 seconds", "switch costume to out", "wait 0.5 seconds", "switch costume to coming out", "wait 0.2 seconds", and "hide". The "else" block also contains a "hide" block. The second script is triggered by "when clicked" and contains a "broadcast 'Spawn a Mole'" block, followed by a "forever" loop. Inside the loop, there is an "if mouse down? and touching color blue?" block. The "if" block contains a sequence of actions: "start sound Boing", "switch costume to hit", "stop other scripts in sprite", "change score by 1", "wait 0.5 seconds", "switch costume to coming out", "wait 0.2 seconds", and "hide". Finally, there is a "broadcast 'Spawn a Mole'" block at the end of the loop. A small mole character is shown in the top right corner of the script area.

Spawn the mole in different holes

Script:

The script is divided into three parts:

- Main Script:**
 - when clicked
 - hide
 - go to x: -132 y: 86
 - create clone of myself
 - go to x: 0 y: 86
 - create clone of myself
 - go to x: 132 y: 86
 - create clone of myself
 - go to x: -132 y: 0
 - create clone of myself
 - go to x: 0 y: 0
 - create clone of myself
 - go to x: 132 y: 0
 - create clone of myself
 - go to x: -132 y: -86
 - create clone of myself
 - go to x: 0 y: -86
 - create clone of myself
 - go to x: 132 y: -86
 - create clone of myself
- When I start as a clone:**
 - show
 - wait 0.5 seconds
 - pick random 1 to 1.5 seconds
 - if pick random 1 to 9 = 1 then go to x: -132 y: 86
 - if pick random 1 to 9 = 2 then go to x: 0 y: 86
 - if pick random 1 to 9 = 3 then go to x: 132 y: 86
 - if pick random 1 to 9 = 4 then go to x: -132 y: 0
 - if pick random 1 to 9 = 5 then go to x: 0 y: 0
 - if pick random 1 to 9 = 6 then go to x: 132 y: 0
 - if pick random 1 to 9 = 7 then go to x: -132 y: -86
 - if pick random 1 to 9 = 8 then go to x: 0 y: -86
 - if pick random 1 to 9 = 9 then go to x: 132 y: -86
- When clicked:**
 - wait 0.5 seconds
 - forever loop:
 - wait pick random 1 to 1.5 seconds
 - if pick random 1 to 9 = 1 then go to x: -132 y: 86
 - if pick random 1 to 9 = 2 then go to x: 0 y: 86
 - if pick random 1 to 9 = 3 then go to x: 132 y: 86
 - if pick random 1 to 9 = 4 then go to x: -132 y: -86

As now all of our holes are created as clones, then how should we spawn our mole in random hole?

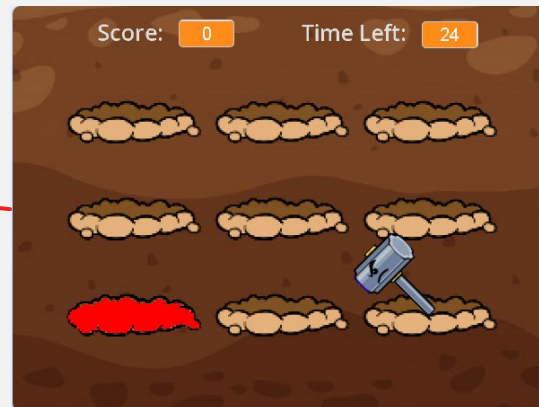
Our can't set a specific number of hole on our clone, therefore one of the ways is to make use of the real hole.

We have hidden our hole at the start, and now we just pick random from 1-9, then move our real hole (hidden) according to the number.

Spawn the mole in different holes

And we program our mole to go to the real (hidden) hole, then now when you run your game you will see your mole can come out with different holes already.

I have marked the real hole in **red color** to show you how the computer execute the program.



Script:

```

when I receive Spawn a Mole
  wait 0.5 seconds
  forever
    wait pick random 1 to 1.5 seconds
    if pick random 0 to 1 = 1 then
      go to hole
      go to front layer
      show
      switch costume to coming out
      wait 0.2 seconds
      switch costume to out
      wait 0.5 seconds
      switch costume to coming out
      wait 0.2 seconds
      hide
    else
      hide
  
```

```

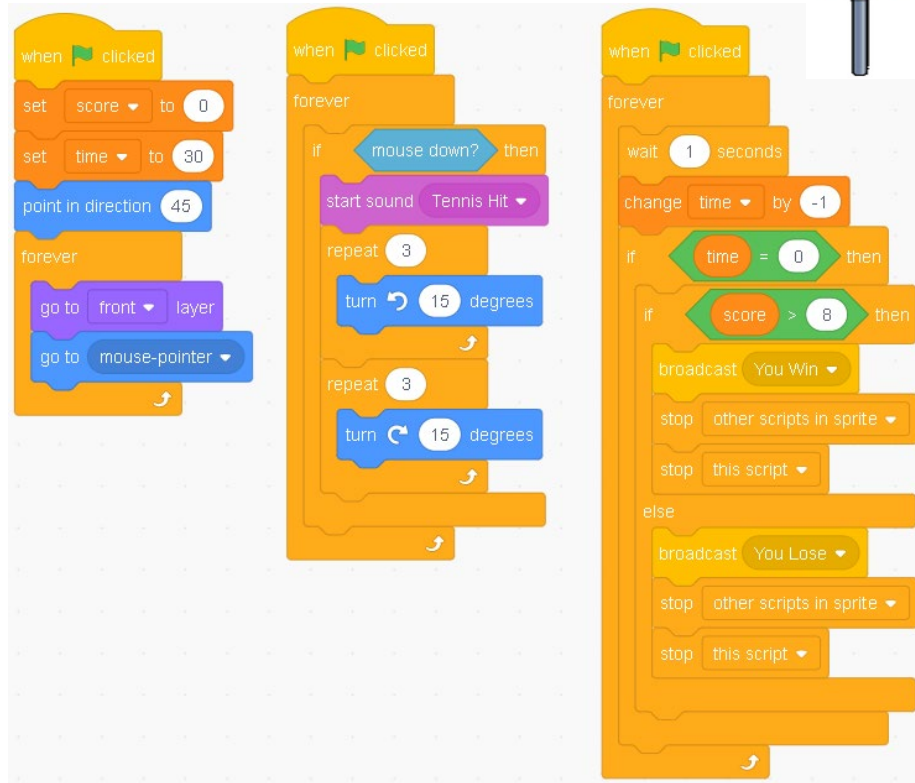
when clicked
  broadcast Spawn a Mole
  forever
    if mouse down? and touching color ? then
      start sound Boing
      switch costume to hit
      stop other scripts in sprite
      change score by 1
      wait 0.5 seconds
      switch costume to coming out
      wait 0.2 seconds
      hide
      broadcast Spawn a Mole
  
```





Program your hammer

Script:



Your hammer got nothing much different than the previous lesson, except the winning score set to “when score > 8” since your winning point is 9 or above.

Do program your win and lose broadcast also, that’s totally same as previous lesson, so I won’t mention again over here.



Solution – Rerun your game and play it



Now you can rerun it and try to play the game.

You will notice that you won't have more than 1 mole coming out at the same time, how should we make it multiple moles coming out at the same time?



Multiple Moles coming out



If we want to allow multiple moles coming out, we will have to make our moles into clones, and we don't need to use the real hole to track for random position anymore.

Setting up clones for moles

Now we want set a clone for each of the holes, so all of these are clones and they have a function to come out from the hole when the clone is created.

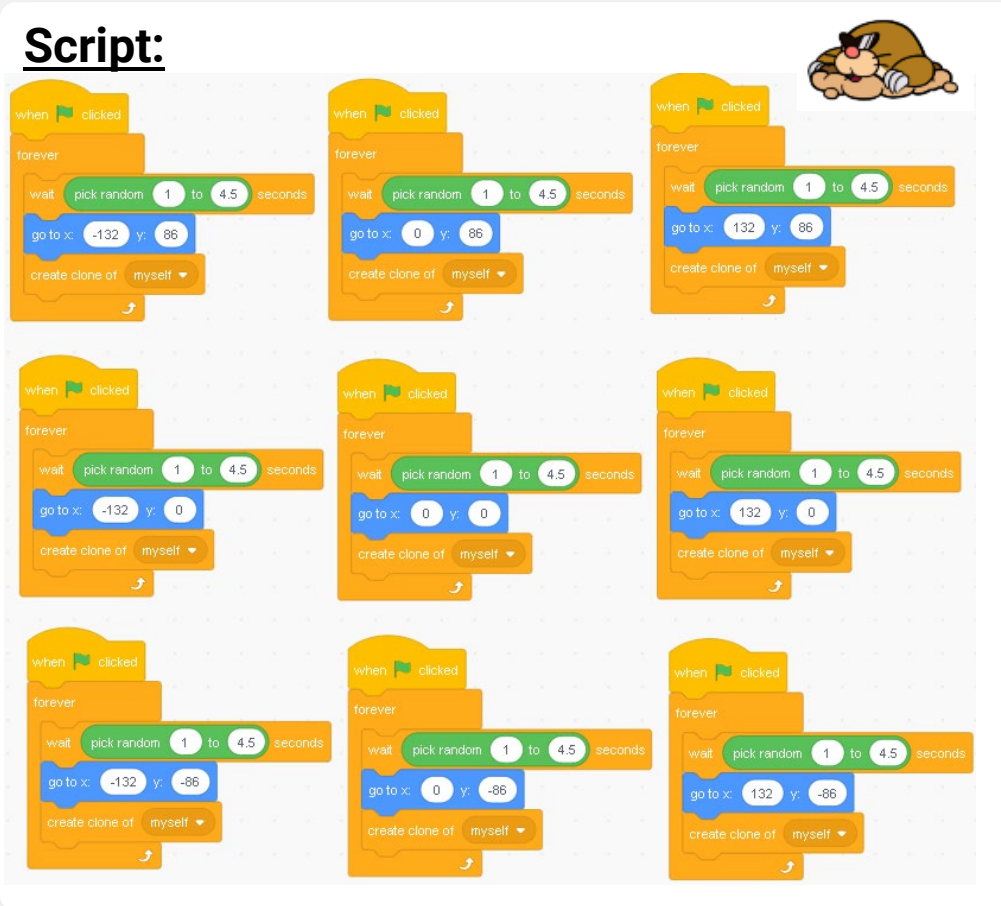
And if the clone get hit by hammer, it will delete itself and add 1 point to the score.





Set up clones for moles (one for each hole)

Script:



Each hole will have 1 mole, so we need to set up 9 moles (clone) running concurrently.

Then we add pick random from 1 to 4.5 seconds of waiting time for the interval of creating clone.



Function of the mole (when start as clone)

Script:



```
when I start as a clone
  go to front layer
  show
  switch costume to coming out
  wait 0.2 seconds
  switch costume to out
  wait 0.5 seconds
  switch costume to coming out
  wait 0.2 seconds
  delete this clone

forever
  if mouse down? and touching color blue? then
    start sound Boing
    switch costume to hit
    change score by 1
    wait 0.5 seconds
    switch costume to coming out
    wait 0.2 seconds
    delete this clone
```

With every 1 to 4.5 seconds (randomly), the clones will spawn at all locations (different time interval for different holes).

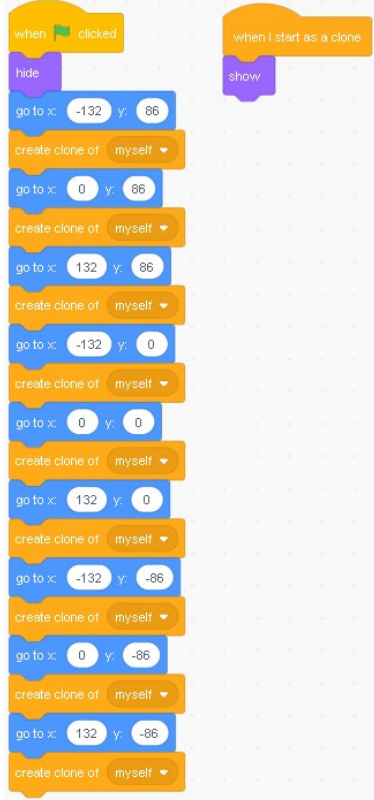
And when the clone is created (spawned), it will go to front layer and show itself, followed by coming out and going back to the hole.

When you hit the clone by hammer, it will delete the clone and add 1 point to the score.



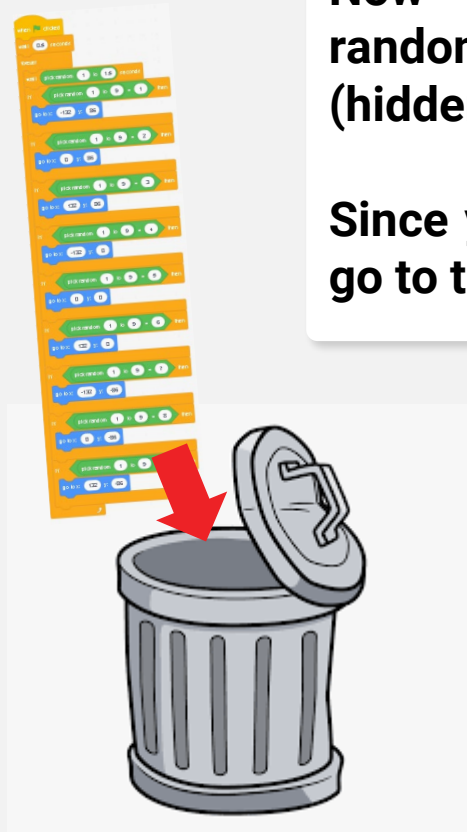
Delete the movement of the real hole

Script:



Now you can throw away the random movement of the real (hidden) hole.

Since you do not need the mole to go to the real hole anymore.



Run and try the game again



Let's give a try to run the game again and see what is different from the last game.

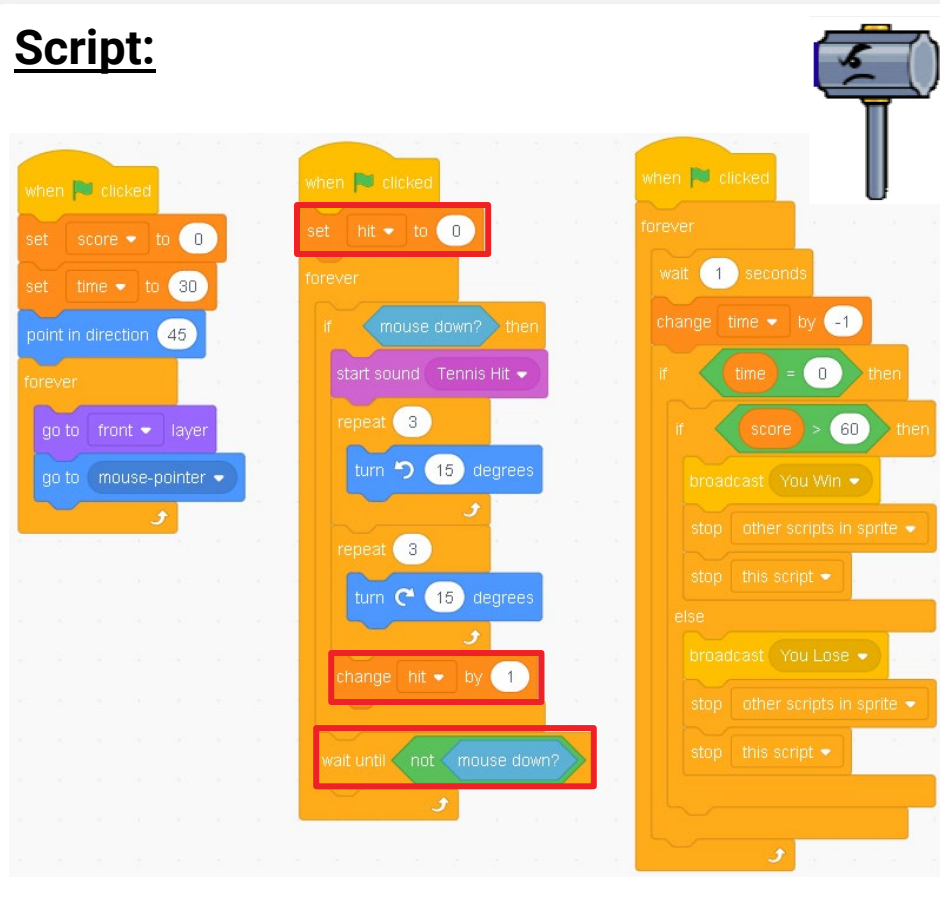
Now you have more moles coming out and it can easily get you score more than 9 points, let's add more rules to the game:

- 1. I also want to measure the success hit rate (accuracy).**
- 2. You will need to score more than 60 points in a game (30 seconds), but with accuracy higher than 90%.**
- 3. We will count how many time you click and how many score you get.**



Measure Accuracy – Trace the Hit count

Script:



Let's set up a new variable to trace the hit count (how many time you click and smash), I will call it "hit" over here.

Then let's set "hit = 0" at the start.

Once we click the mouse, it will turn the hammer for 45 degree counter-clockwise, then turn it back to original point. Then we change hit by 1 (this is counted as 1 hit).

Then I want to make sure that player won't holding the mouse down forever, so I will set a wait until mouse is up (not mouse down means mouse up), then the hitting action will be enabled again.

Show the hit variable



Now let's show up the variable "hit" on the screen, and make a label (the "Hit: " word) on the left side of the readout.

So that we can see how many time we hit and how many score we get.

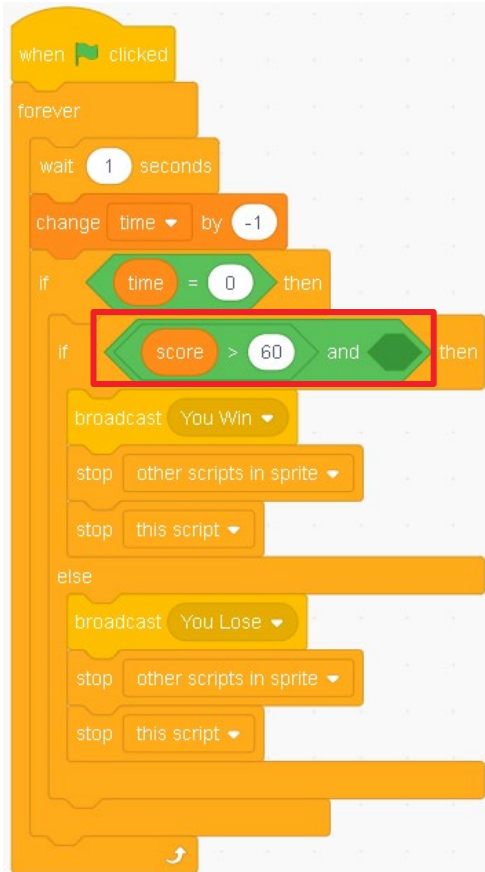
So the more you hit but less score you get means lower accuracy.

The closer the count of hit and the score, the more accurate you are.



Defining winning condition

Script:



Now we want to set the wining condition.

Player must get more than 60 scores and with accuracy more than 90%.

This means that if you hit 100 times, you must score at least 91 (means you can't miss the hit for 9 times).

Either you failed to achieve one of these (score > 60, accuracy > 90), you lose, else you win.

Let's put an "and" block to it to define the winning condition.



Formula - Calculate the Accuracy

$$\text{Exam Marks} = \frac{60}{100} \times 100\% = 60\%$$

$$\text{Exam Marks} = \frac{28}{50} \times 100\% = 56\%$$

$$\text{Accuracy} = \frac{\text{Score}}{\text{Hit}} \times 100\%$$

$$\text{Accuracy} = \frac{65}{70} \times 100\% = 92.8\%$$

Let's take your exam marks as an example to do the calculation.

If total marks is 100, and you only get 60, how many percent you score in your exam? Ans = 60%

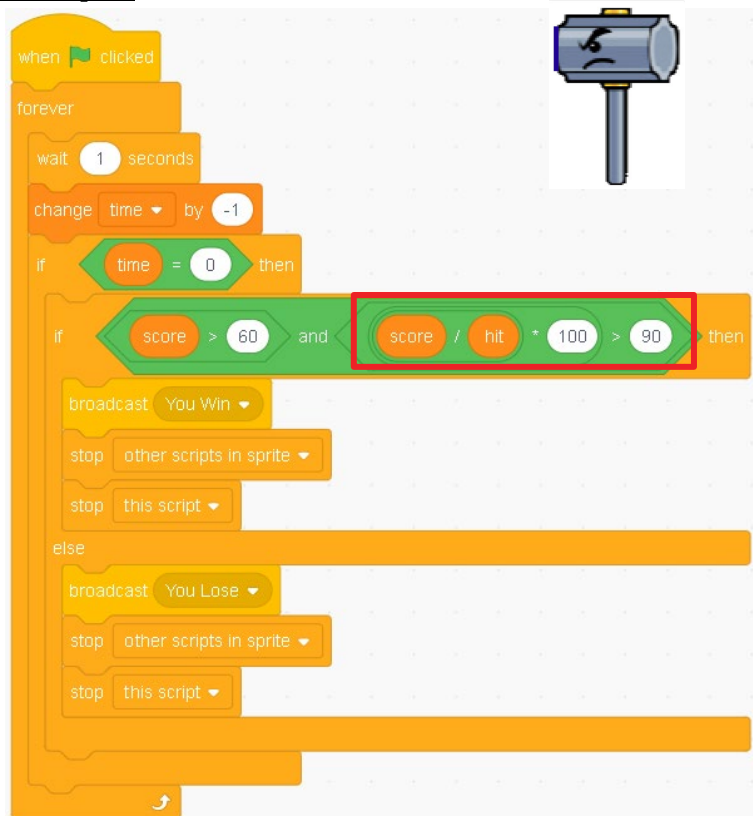
If full mark is 50, you score 28, means you get 56% out of 100%.

Same to accuracy, if you hit 70 times, and only 65 success hit, you can use 65 divide by 70 then times 100%, then you get around 92.8%



Formula - Calculate the Accuracy

Script:



$$Accuracy = \frac{Score}{Hit} \times 100\%$$

Let's use this formula to calculate our accuracy.

$$(Score / hit) * 100 > 90$$



Let's run and play the game



Now let's run and play the game again.



ASSIGNMENT *for*

Lesson 2-2



L2-2 – Mission

Let's show the accuracy at the end of the game, no matter you win or lose.

Can you try to do a retry button at the end too?

Player can click the retry button and start the game again.



You can direct message your teacher and ask your question through [Slack Robotene Community](#) or arrange a [One-to-One Consultation](#) with your teacher.



Any Questions?



Thank you :)