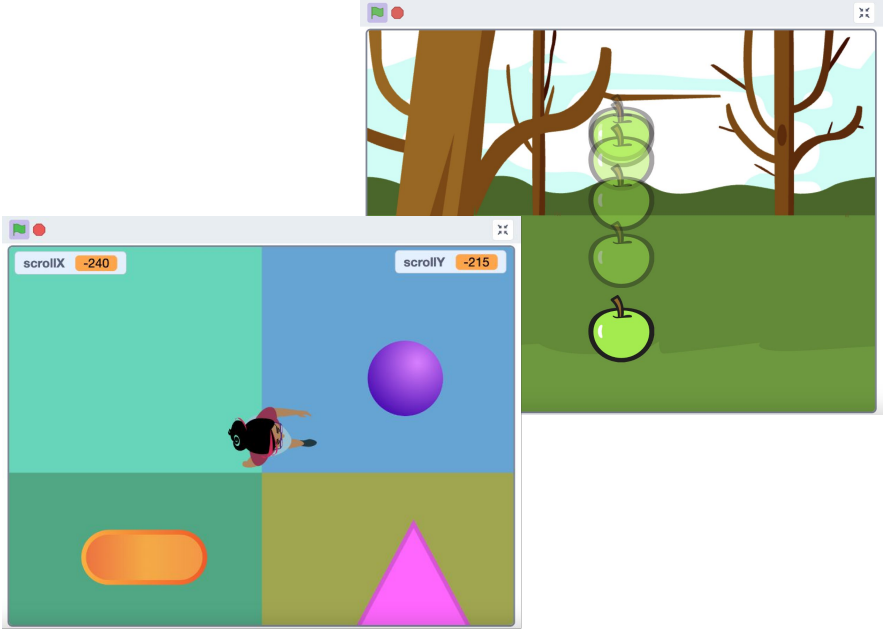
Advanced Topics: Graphic Effects

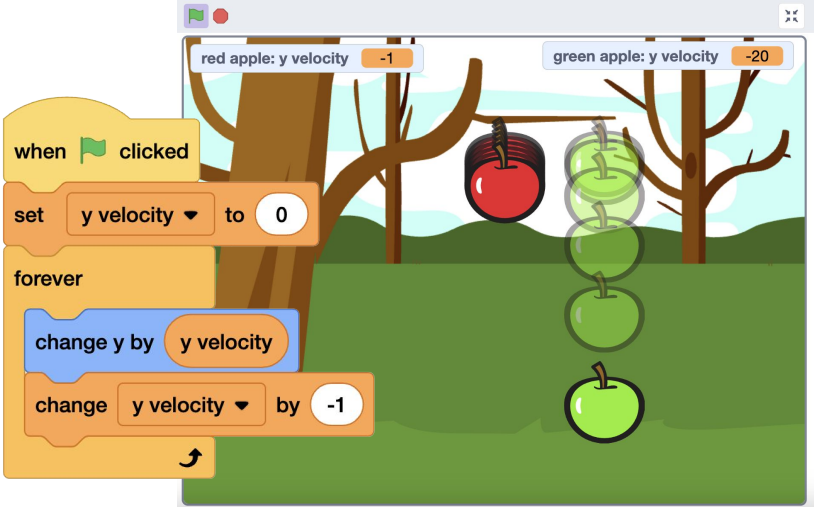


Use data to create more advanced scripts for games, stories, and more!

Cards in This Pack

- Gravity in Scratch
- Gravity in Scratch: Stop or Bounce
- Gravity in Scratch: Reverse Gravity
- User-Controlled Scrolling Background
- Text Generator/Text Rendering [COMING SOON]

Gravity in Scratch



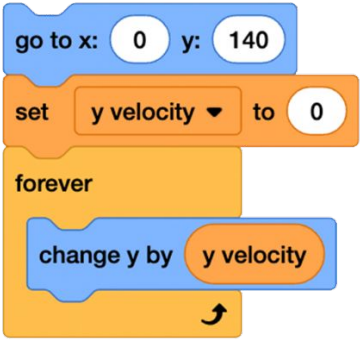
Sir Isaac Newton is said to have discovered universal gravitation by observing the fall of an apple.

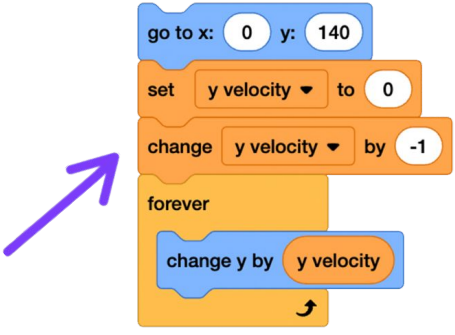
Gravity in Scratch means sprites fall to the bottom of the stage in a realistic way, slowly gaining speed as they fall. Velocity is the speed of something in a given direction.

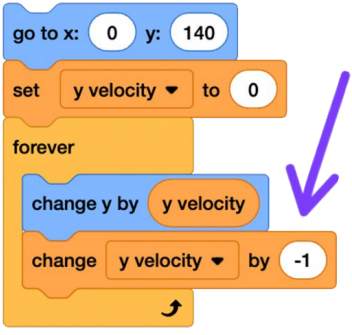
Gravity in Scratch

scratch.mit.edu

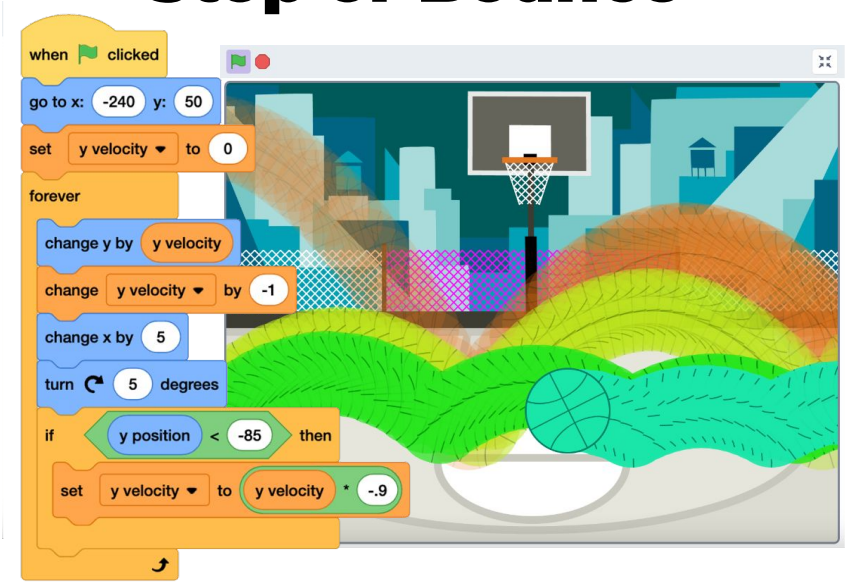
1. To begin creating gravity, set up a variable that will hold the velocity of your sprite, like “y velocity.”
2. Next, set the velocity to zero at the start (since the object will start from a place of rest).
3. Then, have the sprite move (in this case, change y) by that velocity.
4. In order to gain speed (acceleration), the velocity will have to change. what is difference if that change happens inside or outside your loop?







Gravity in Scratch: Stop or Bounce



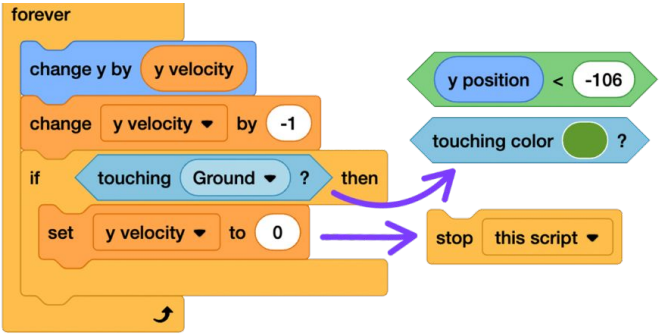
When creating gravity in Scratch, you might also think about how the object should react when it hits the ground.

The object could come to a complete stop. Or sprites may bounce when they reach the ground (or the bottom of the stage), with those bounces getting smaller and smaller over time.

Gravity: Stop or Bounce

scratch.mit.edu

1. To stop the sprite from falling through to the bottom of the stage, you'll need to set up a condition that tells the program when to stop the sprite. You can choose from many conditions, like the position, touching a sprite, etc.



2. When the condition has been met, you could stop the script completely or stop the movement by making the velocity 0, back to a place of rest.

3. In an inelastic or "sticky" collision, the kinetic energy is absorbed by its surroundings and the object comes to an immediate stop. In an elastic or "bouncy" collision the kinetic energy (and thus the bounces) gets smaller over time. Try to recreate this and experiment by setting the velocity to the velocity times numbers like -1 or - 0.9.

4. What if you wanted to also move an object along the other axis as it fell or bounced? (See the card front.)

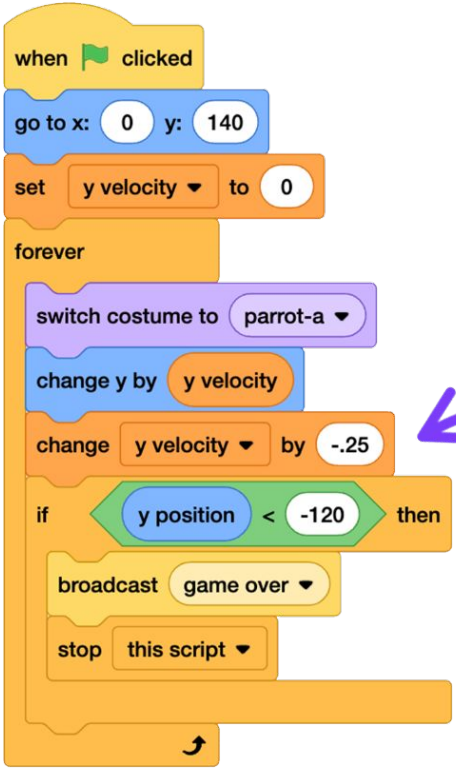
Gravity in Scratch: Reverse Gravity

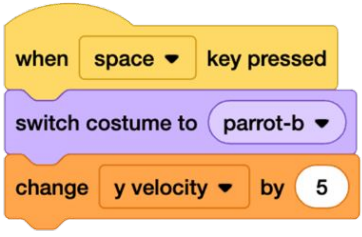
What if you wanted to, at least temporarily, reverse gravity? One example is in the case of a “flappy bird”-style game, where pressing a keyboard key temporarily reverses gravity and stops a sprite from hitting other objects or the ground in order complete the objectives of a game.

Gravity: Reverse Gravity

scratch.mit.edu

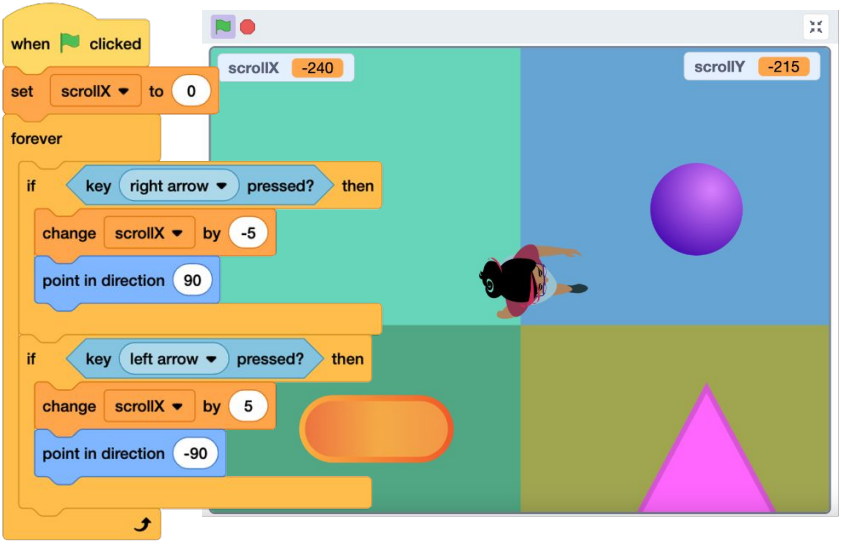
1. Create a script that sets your sprite to fall with gravity.
2. Then, add a second script so when, for instance, a keyboard key is pressed, the velocity will change by a larger positive number. (Versus the smaller negative number in the gravity script.)





Test and experiment with the numbers. For a moment, the velocity should be smaller, becoming positive if you hit the key enough times, momentarily slowing or reversing the gravity effect.

User-Controlled Scrolling Background



You can create a version of a scrolling background that moves behind your sprite as your sprite stays in place.

Your user can control the scene with keyboard keys to move through it and explore. Scrolling backgrounds like this could be used in games or animations or informational projects.

User-Controlled Scrolling Background

scratch.mit.edu

GET READY

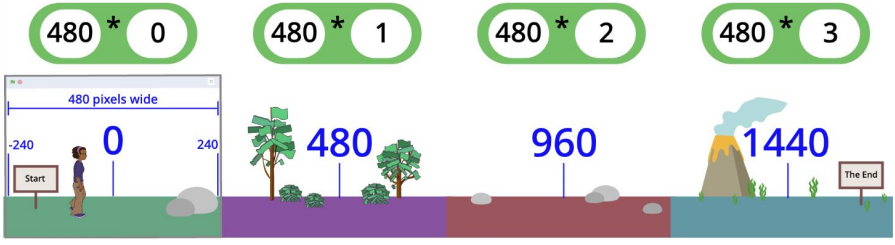
Choose a sprite, perhaps with costumes to show movement.

Draw a few backgrounds as sprites.

ADD CODE

1. To line up all the background sprites end-to-end, knowing that the stage is 480 pixels wide, you can use a mathematical expression to quickly position each sprite. Set the x-position of the first background at 480 times zero (480 x 0 = 0). The next at 480 times one, etc.

2. Now, create a variable that will allow you to change the position of all the background sprites simultaneously.



(Continue to the next card.)

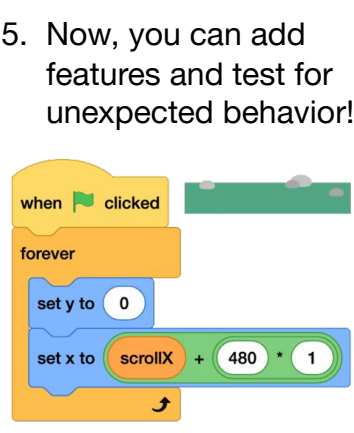
User-Controlled Scrolling Background

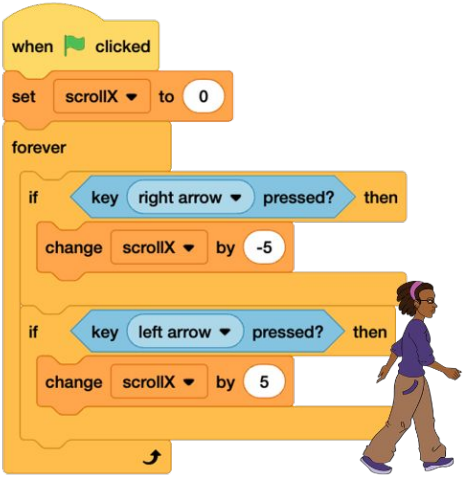
scratch.mit.edu

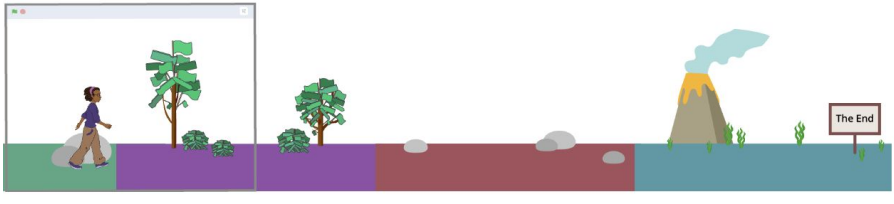
3. On the main walking sprite, create a script that changes the variable (scrollX) by a positive or negative amount when keys are pressed.

4. Then, on the background sprites, adjust the x or y position to add the variable.

5. Now, you can add features and test for unexpected behavior!







User-Controlled Scrolling Background

scratch.mit.edu

EXPERIMENT!

- What if you want the sprite to look like it is walking by cycling through the costumes?
- What if you want to ensure that the walking sprite can't go off the edge of the first or last background sprite?
- What if you want to add other sprites to interact with your walking sprite?
- What if you want to move your sprite along the x and the y axis?