

Game Design

Become A Game Designer

By: Seven Hu

Overview:

In the previous lesson, we've learned how to design game rules. Today, we are going to spice up our games by increasing the difficulties of the game.

In the game we've designed last time, the track is fixed there. But this time, we are going to increase the difficulties by designing moving obstacles in the game. Moreover, the place where the moving obstacle appears keeps changing, which means the obstacle will come out of nowhere each time.

What's upgraded in the rules:

- Game rule:

The racing car has to avoid the bats falling from the sky when driving.

- Outcomes:

The game will be over once the bat hits the bat;

The game continues if the car doesn't hit the bat. But the bat will appear again later.

Questions:

- How do we make the bat keep falling from the sky?
- How do we write programs to ensure that the bat explodes once it hits the car? (switch the sprites)
- How do we make the bat appear in the sky again after it hits the ground?
- How do we make the bat come out of a random place each time?

Brainstorming

- Have students discuss those questions in groups and list their ideas on the blackboard. In today's lesson, we are going to learn about two new blocks:



The bat falls from a random place.



The bat falls from somewhere in the sky again after it hits the ground.

Prototyping:

- Physical prototype: The teacher draws a bat as the draft.
(Bat 1: A complete bat;
Bat 2: A bat smashed to pieces by the racing car.)
- Digital prototype: Use mBlock 5 to code the bats. First add the sprite “bat” to the stage.

Playtesting

- ① How does the bat know the game has begun?

Codey



Bat



Playtesting

- ② How do we make the bat come out of a place randomly and unexpectedly each time?



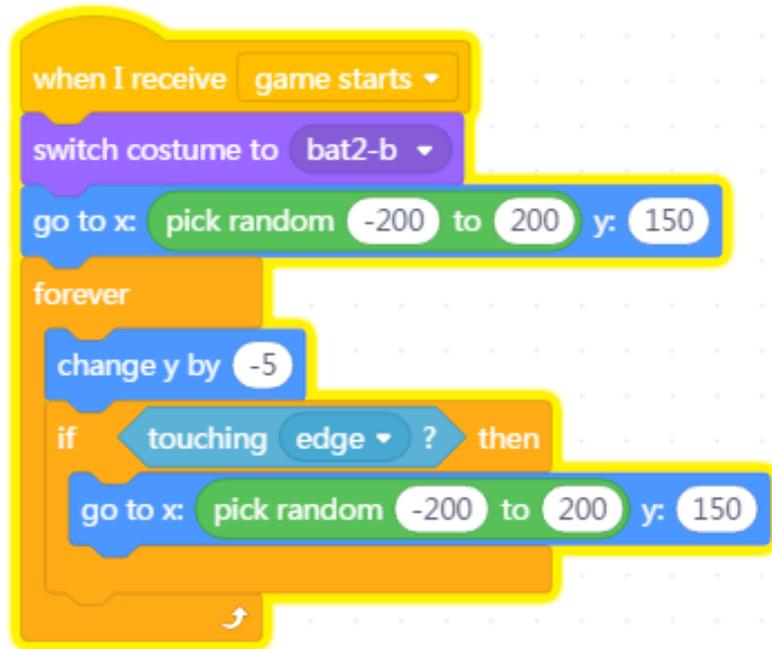
Playtesting

- ③ How do we keep the bat falling from the sky?



Playtesting

- ④ How do we make the bat appear in the sky again after it hits the ground?



Playtesting

- ⑤ How do we write programs to ensure that the bat explodes once it hits the car? (switch the sprites)

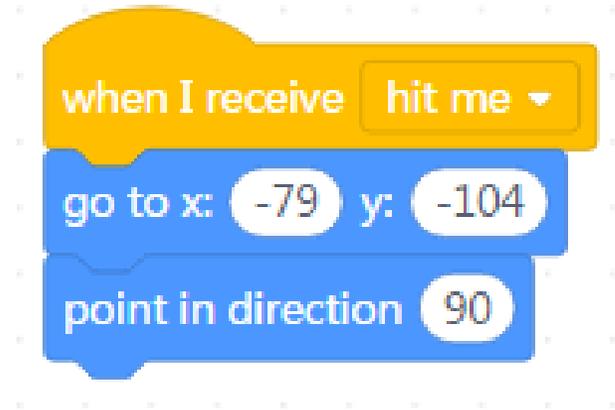


```
if touching Car1 ? then
  switch costume to crashed bat
  broadcast hit me
  wait 1 seconds
  switch costume to bat2-b
  go to x: pick random -200 to 200 y: 165
```

The image shows a Scratch code block with the following steps: 1. An 'if touching Car1 ? then' block. 2. A 'switch costume to crashed bat' block. 3. A 'broadcast hit me' block. 4. A 'wait 1 seconds' block. 5. A 'switch costume to bat2-b' block. 6. A 'go to x: pick random -200 to 200 y: 165' block.

Playtesting

- ⑤ How do we make the racing car go back to the place where it sets out after it's hit by a bat?

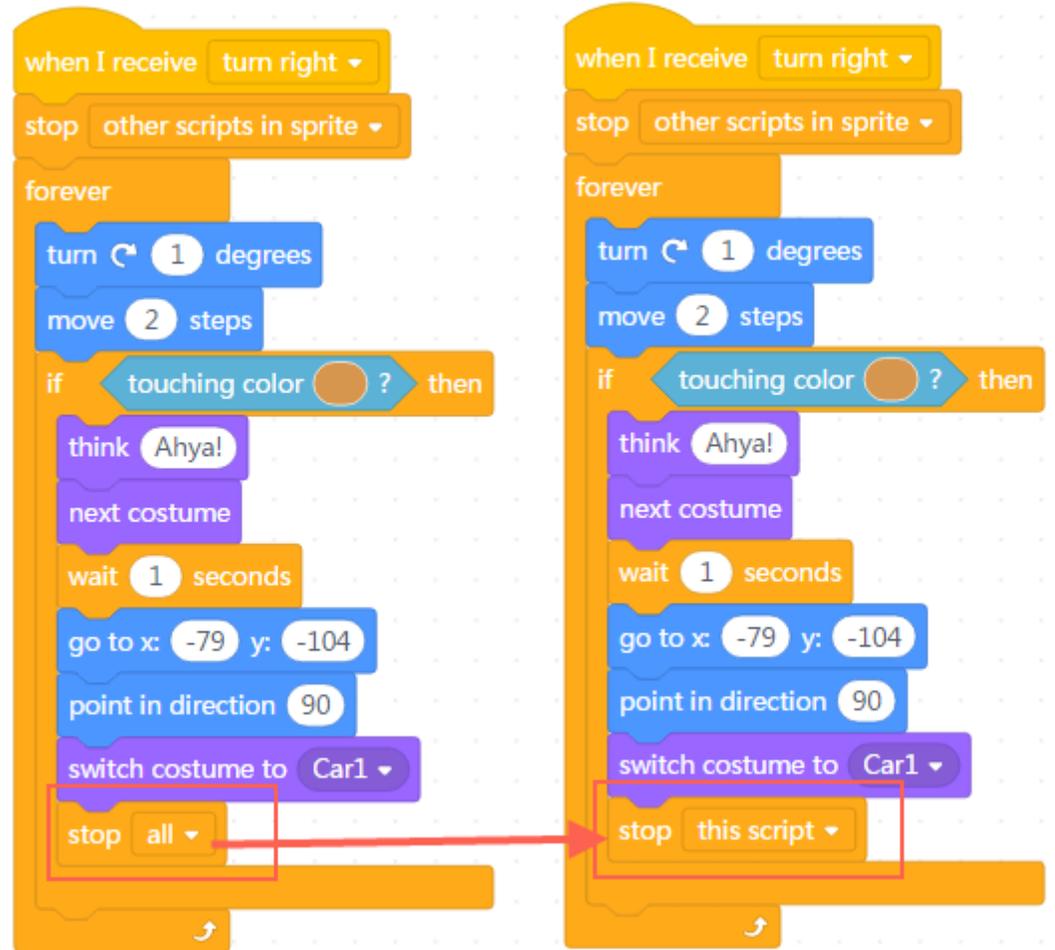


Reiterating & Implementing:

- **What's the problem: Why does the bat come to a halt all of a sudden?**

Reiterating & Implementing:

- **Identify the issue:** The program for controlling the racing car needs revising:
- What's the difference between the two blocks:



Independent Practice

- **It's Your Turn!**
- **1)Brainstorming:** “We've just discussed how to make your racing game more engaging. You came up with so many ideas, but now you have to narrow down the ideas to only one. After that, you can start creating your game.”
- **2)Prototyping:** “Add sprites to your game based on your design concept. Then starts coding the sprites.”
- **3)Playtesting:** “When your game is ready, put your hands up. You can invite target players to playtest your prototype. Of course, you can invite me to try it.”
- **4)Reiterating & Implementing:** “Fix bugs based on the players' feedback. Try everything to perfect your game.”

Share

- **Presentation:** Encourage each student to share his or her own game with the class and tell them to invite classmates to experience their game.
- At the end of this session, let students vote for the **Best Game of the Day**.

Extensions

- Add a new moving obstacle (gold coin).
- As defined by the rules, the racing car should capture the gold coin before the coin falls on the ground and tries to avoid the bats at the same time.