12

SPACE WARS

The Game Plan:

Let's make a Space Wars Game with a spaceship shooting missiles at attacking enemy craft. Each missile hit will add a score but each enemy craft that gets through will subtract a score. A count down timer will add more excitement with the possibility of creating a top score competition that you can play with your friends. Let's start!

The Stage:

1. Import the **'stars'** background from the **Nature** folder.

2. Select the **'Variables'** section and create two new variables, **'Time'** and **'Score'**. Set these variables to **'for all sprites'**.

3. Add the following code to the Stage script. This code will make the Time variable count down from a number of your choice, to zero. If it reaches zero, the game will end.

TEST 1:

When the green flag is clicked the score is set to zero and the countdown time is set to one minute.

Countdown begins immediately. Do the monitors show that the variables are functioning correctly?

Now we're going to make the scripts, make 4 Sprites called Spaceship, Enemy Craft, Enemy Craft 2 and Missile. Leave all of the costumes black, we'll make these in each section.
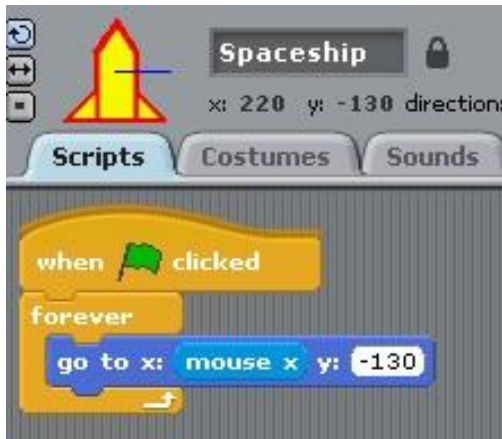
The Space Ship:

4. Draw a spaceship. You can keep it simple like the one in the image below.
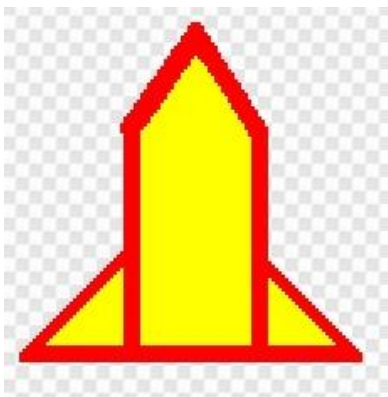
5. Program the spaceship with the script below. This code will keep your space ship at the bottom of your screen, but will allow it to follow your mouse from left to right.

TEST 2:

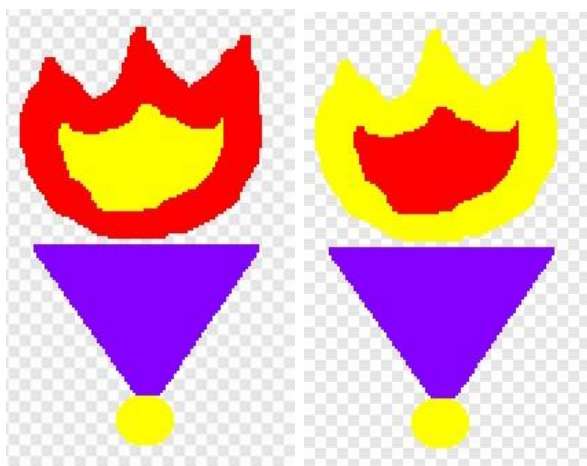Does the spaceship follow the mouse in a straight line across the bottom of the stage?

Challenge: What other ways could you program the spaceship to achieve the same, or a better effect?

The Enemy Craft:

6. Draw the enemy craft. Create two costumes to animate the flame trail and create a flashing effect. Create a few other costumes for when the enemy craft gets hit by the missile. These will give the effect that the enemy craft is exploding!
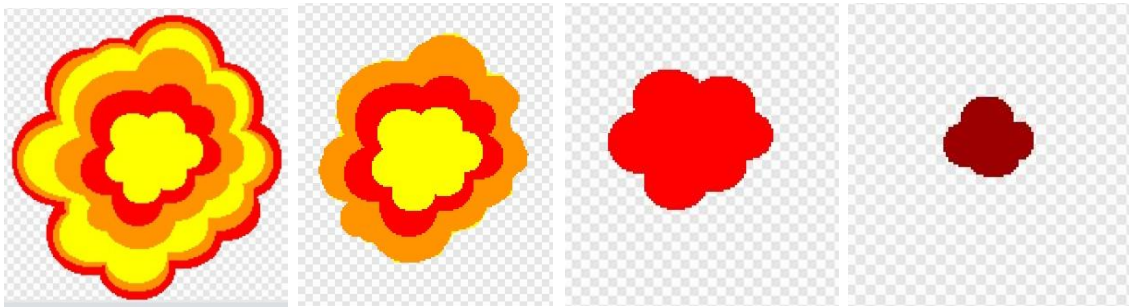
Tip: The costumes for the explosion in the diagrams below are simply made by drawing circles inside other circles, using just red, yellow and orange! Each new costume for the explosion is smaller than the previous costume, making it look like the explosion is fading away!

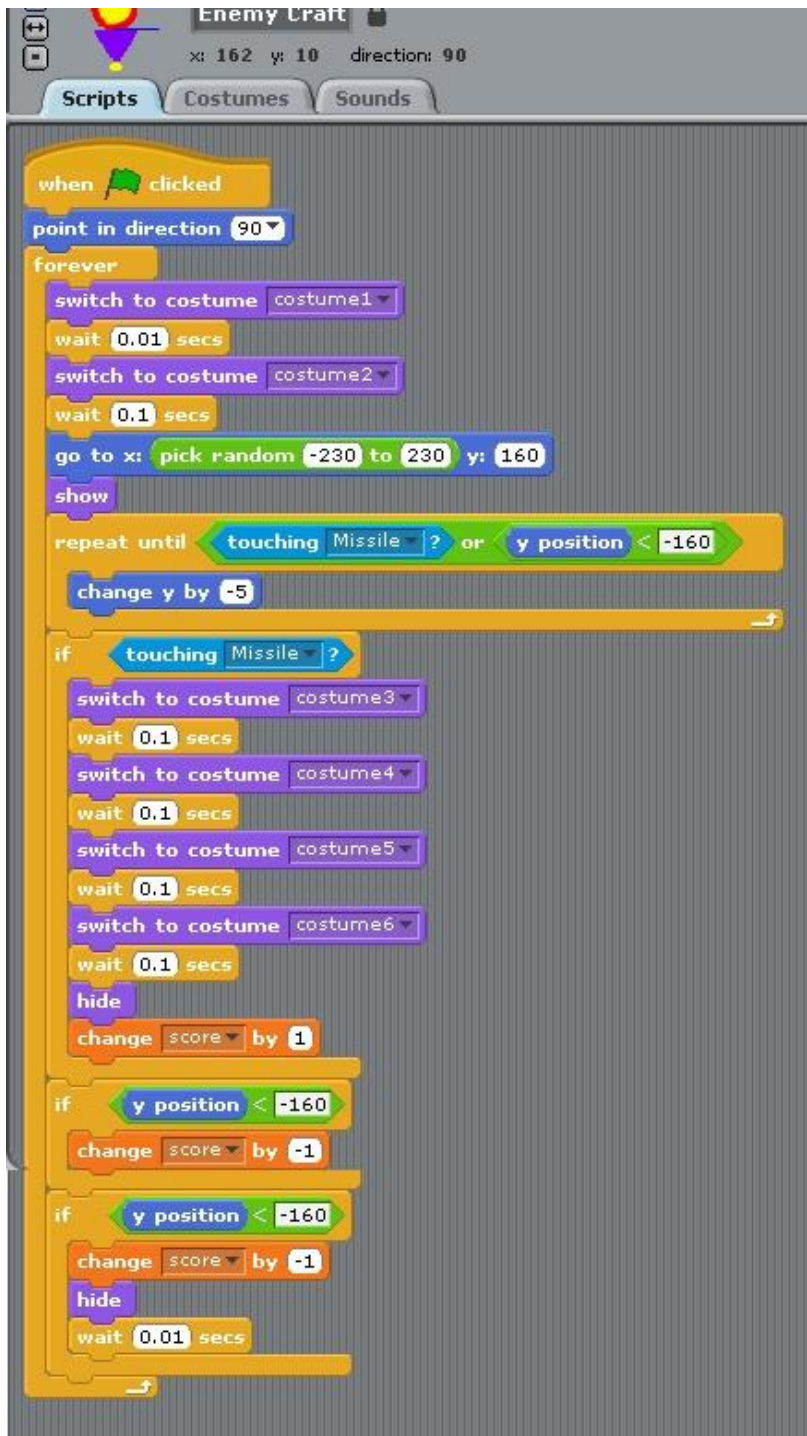Enemy craft costumes 1 and 2:

Enemy craft costumes 3, 4, 5, and 6:

TEST 3:

Do the enemy craft appear and disappear randomly when it is hit by a missile or when it reaches the bottom of the stage?

7. Now program the enemy craft with the script below. This code includes the following instructions:

- It tells the enemy craft to change between costume1 and costume2 to make it look like it has a flaming trail.

-It tells the enemy craft to move down the screen (by using the **'change y to'** block).

- It tells the enemy craft to switch to costumes 3, 4, 5, and 6 and then hide if it gets hit by a missile.

- If it gets hit by a missile or reaches the bottom of the screen, the enemy craft will hide for a moment before returning back at the top of the screen, at a random location (by using the **'go to x and y'** block and the **'pick random -230 to 230'** block).

- If the enemy craft reaches the bottom of the screen without getting hit, the player gets deducted 1 point (by using the **'change score by -1'** block).

12

Enemy Craft
x: 162   y: 10   direction: 90

| Scripts | Costumes | Sounds |

```
when [green flag] clicked
point in direction (90▼)
forever
    switch to costume [costume1▼]
    wait (0.01) secs
    switch to costume [costume2▼]
    wait (0.1) secs
    go to x: (pick random (-230) to (230)) y: (160)
    show
    repeat until <<touching [Missile▼]?> or <(y position) < (-160)>>
        change y by (-5)
    if <touching [Missile▼]?>
        switch to costume [costume3▼]
        wait (0.1) secs
        switch to costume [costume4▼]
        wait (0.1) secs
        switch to costume [costume5▼]
        wait (0.1) secs
        switch to costume [costume6▼]
        wait (0.1) secs
        hide
        change [score▼] by (1)
    if <(y position) < (-160)>
        change [score▼] by (-1)
    if <(y position) < (-160)>
        change [score▼] by (-1)
        hide
        wait (0.01) secs
```

12

The Missile:

8. Draw the missile. Something like the one in the diagram below will be perfect.

9. Program the missile with the script below. This code will place the missile wherever your spaceship is when you press the space bar. It will then shoot upwards until it either hits the enemy craft, or it reaches the edge of your screen. This is what the **'change y by 30'** block does. The '30' is the speed at which the missile moves. You can change this to go as fast or as slow as you like!

NOTE: The missile is activated by pressing the space bar. It's unlike the other sprites and the score variable, which are activated when the green flag is clicked. You have to be ready to hit the space key immediately or you could easily lose scores and waste valuable seconds!


TEST 4:

Does the spaceship fire a missile upwards when you hit the space key? Is the missile fast enough for repeated, rapid fire?

12

We're nearly there! Now all we have to do is create two new sprites. One will tell the player if he or she wins the game. The second one will tell the player if he or she has lost the game.

10. Select **'Paint new sprite',** select the text option (the **'T'** icon) and write your message. I have written "You win!!!" and put a black box around it.
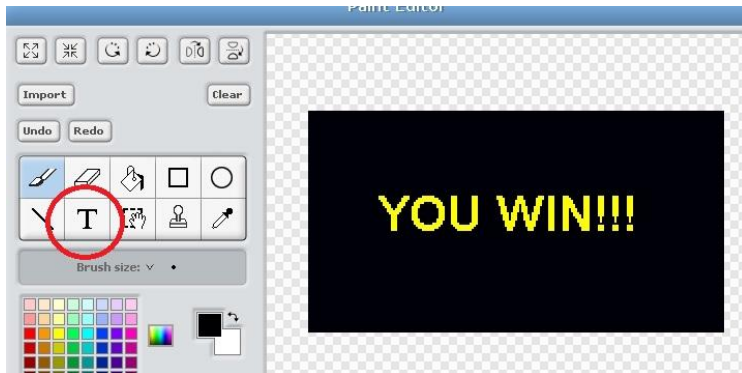
11. Now add the following code to the script for the "You win!" sprite. This code will make the sprite hide the sprite from the screen unless the player reaches the required score. I have set the score to 50 but you can set it to whatever number you like.

12. Now repeat step 8, except this time write a message that tells the player they've lost the game. I've written "Game over!!!"

13. Add the following code to the "Game over!" sprite. This code will make the sprite appear on the screen if the time runs out before the player reaches the required score.

12

TEST 5:

Does the "Game over!" sprite appear when the time reaches zero?

WELL DONE!!! You've now finished this exercise! Can you think of any other ways to make our game more exciting? Have fun playing it!!