Meteor Dodge – Creation Tutorial

https://scratch.mit.edu/projects/67076150/

Paint a **new sprite**

- This will be our template space background
- Use the paint tools under the costume tab to draw a black square that fills the entire canvas
- Draw a small white circle and duplicate it many times to look like stars
- Make sure you have chosen *Vector Mode*!
- Name this sprite space0_0



Create the following **variables** using the *Make a Variable* under the **Data** blocks section

- We use capital letters on variables to represent constants (these do not change their value)
- We don't need to see all these variables on-screen except for shipLives and timer

```
METEOR_NEG_SPEED

METEOR_POS_SPEED

SHIP_NEG_SPEED

SHIP_POS_SPEED

difficulty

scrollX

shipLives

timer
```

Script **space0_0** sprite with the following scriptlet:

- This is how the scrolling works in the game
- We move the background object rather than the player object itself

```
when clicked

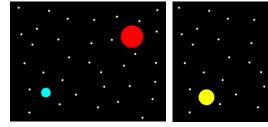
forever

go back 1 layers

go to x: scrollX + 480 * 0 y: 0
```

Duplicate the **space0_0** object 4 times (to have a total of 5 **spaceX Y** sprites).

- Name each of these as space1_0, space2_0, space-1_0, space-2_0
- On space1_0 and space-1_0, draw some coloured circles to represent suns



On each of the *spaceX_Y* sprites, change the number in the (*scrollX* + 480 * spaceX) scriptlet to whatever the **X** value in the sprites name is, e.g

 space-1_0 will have a -1 and space2_0 will have a 2

```
when clicked

forever

go back 1 layers

go to x: scrollX + 480 * -1 y: 0
```

Script the **Stage** with the following scriptlet

- We use three broadcast calls to assign values to our variables we created earlier, show the start game message, and start the game
- We forever check to see if our scroll is greater than or less than the center of our outside spaceX_Ysprites (i.e. space-2_0 and space2_0). When we get past these, we reset scrollX so that we move back to space0_0. This is why our edge backdrops must match our center backdrop (i.e. space-2_0, space2_0 and space0_0 all look the same)
- We also forever check if the game is over by checking our shipLives and if so, broadcast call the end game message

Script the **Stage** with the following scriptlet:

- This sets the initial values of all our variables and hide the variables we don't need to see
- Our variables in capital letters are constants that will not change during game
- SHIP_POS_SPEED and SHIP_NEG_SPEED control how fast our player ship moves left and right
- METEOR_POS_SPEED and METEOR_NEG_SPEED control how fast the meteors move left and right

```
when I receive initVariables v

set difficulty v to 2

set scrollX v to 0

hide variable scrollX v

set shipLives v to 3

set timer v to 0

hide variable timer v

set SHIP_POS_SPEED v to 4

hide variable SHIP_POS_SPEED v

set SHIP_NEG_SPEED v to -4

hide variable SHIP_NEG_SPEED v

set METEOR_POS_SPEED v to 10

hide variable METEOR_POS_SPEED v

set METEOR_NEG_SPEED v to -10

hide variable METEOR_NEG_SPEED v
```

Script the **Stage** with the following scriptlet:

• This starts the timer that we use to judge how well the player has scored

```
when I receive startGame 
forever

wait 1 secs
change timer by 1
```

Create a **new sprite** (from the sprite library)

- Select Planet2
- Name this sprite meteor



Planet2

Script the **meteor** sprite with the following scriptlet:

- This sets the initial size and starts the cloning
- We control how may meteors are fired every second by the *difficulty* variable
- We launch the batches of meteor(s) every 2 to 4 seconds

Create a **new sprite** (from the sprite library)

- Select Spaceship
- Name this sprite *spaceship*



Go back to the **meteor** sprite and script with the following scriptlet:

- We use the **More Blocks Make a Block** and name it *shootMeteorRight*
- This controls the behaviour of the meteors travelling right

```
define shootMeteorRight

show

go to x: -227 y: pick random -155 to 155

repeat until touching spaceship * ? or touching edge * ?

turn * A5 degrees

change x by METEOR_POS_SPEED

delete this clone
```

Script the **meteor** sprite with the following scriptlet:

- We use the **More Blocks Make a Block** and name it *shootMeteorLeft*
- This controls the behaviour of the meteors travelling left

```
define shootMeteorLeft

show

go to x: 227 y: pick random -155 to 155

repeat until touching spaceship * ? or touching edge * ?

turn (* 45 degrees
change x by METEOR_NEG_SPEED

delete this clone
```

Finally, script the **meteor** sprite with the following scriptlet:

 This will control which way the meteor travels depending on which arrow key is pressed (and when neither key is pressed)

```
when I start as a clone

if key left arrow pressed? then

shootMeteorRight

if key right arrow pressed? then

shootMeteorLeft

shootMeteorRight
```

Script the **spaceship** with the following scriptlet:

- This sets where the player sprite starts when the game starts and the size of the sprite
- We also set the player controls for up, down, left, and right
- Left and Right allow the ship to move a little before scrolling across the different background sprites
- Up and Down allow the ship to reach near the top and bottom of the screen

```
when I receive startGame *
go to x: 0 y: 0
  go to front
  set size to 30 %
  if key up arrow pressed? then
    point in direction 90▼
          y position | < 160 | then
      change y by SHIP_POS_SPEED
     key down arrow ▼ pressed? > then
    point in direction -90▼
        y position > -160 then
      change y by SHIP_NEG_SPEED
     key left arrow ▼ pressed? then
    point in direction 0
        x position < -50 then
      change scrollX ▼ by SHIP_POS_SPEED
      change x by SHIP_NEG_SPEED
      key right arrow ▼ pressed? then
    point in direction (1807)
       x position > 50 then
      change scrollX ▼ by SHIP_NEG_SPEED
      change x by SHIP_POS_SPEED
```

Finally, script the **spaceship** with the following scriptlets:

- This sets what happens when a meteor hits the spaceship
- Also hides the ship at the very start and end of the game (where the messages appear)

```
when I receive startGame forever

if touching meteor ? then

change shipLives by e1

when clicked

hide

when I receive endGame hide
```

Paint a new sprite

- Add some text that looks like a game start message
- Name this sprite *startGameMessage*



Script the **startGameMessage** sprite with the following scriptlet:

 This shows the start game message and waits for the player to set the difficulty of the game



Paint a **new sprite**

- Add some text that looks like a game end message
- Name this sprite *endGameMessage*



Script the **endGameMessage** sprite with the following scriptlet:

- This shows the end game message
- If the timer variable is not where you want it to be, simply move it into the place where you want and then replay the game

```
when I receive endGame v when clicked show hide show variable timer v stop all v
```