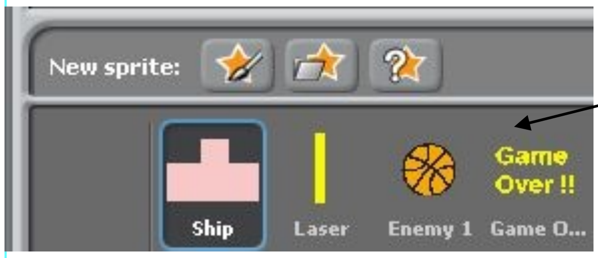
How to Build A Shooter Game using SCRATCH (Advanced)

In these lessons I will show you how to build a basic shooter game using the SCRATCH programming language. The purpose of these lessons is to show you how to program a game and not how to make the "coolest looking or most interesting game" (that's up to you). This is an advanced lesson and assumes you are familiar and comfortable with SCRATCH and programming concepts.

How the Game Works

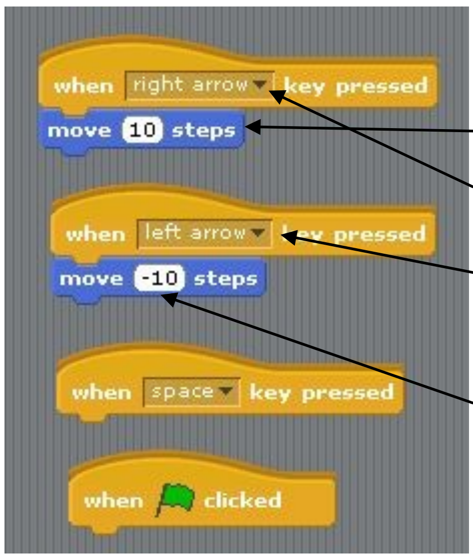
When the green flag is clicked enemies (in my game basketballs) randomly fall from the sky. A variation you could easily add would be enemy ships running patterns in random order. You control a base ship that moves back and forth with the arrow keys on the bottom and uses the space bar to fire a laser. If the laser hits the enemies you score a point. If the enemies fall on you before you shoot them you loose a life. If you loose three lives the game is over. There are sounds when you fire your laser or are hit by an enemy ship. So with that let's get started.



Getting Started

The first thing you will need to do is to create four sprites for your game and name them "*Ship, Laser, Enemy 1 and Game Over*". You may choose different names for your characters but you will need to make some changes to these instructions if you do.

We will start by getting our ship moving across the screen side to side and setting it up to fire the laser.



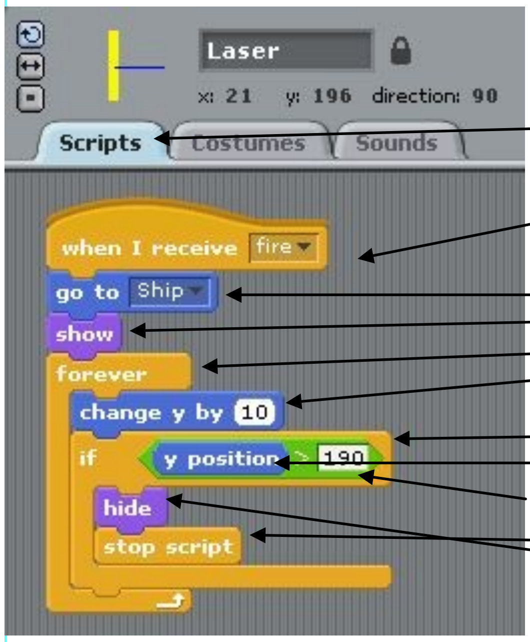
```
when right arrow key pressed
move 10 steps

when left arrow key pressed
move -10 steps

when space key pressed

when 🟩 clicked
```

In the ship's script area, select four top hat blocks from the **Control block** three that allow you to assign keys and one that allows you to change things when the flag is clicked. Now from the **Motion Block** select a Move 10 Steps block and attach it to a key control block and set the key control block to a right arrow key. This will allow you to move your ship right with the right arrow key.

Now from the **Motion Block** select another Move 10 Steps block and attach it to a key control block and set it to a left arrow key. Now change the value in this Move 10 Steps block to –10. This will allow your ship to move left with the left arrow key (as you take away 10 spaces).

Place your ship at the bottom of the stage and test your program by tapping on the arrow key's. Your ship should move right and left.

Setting up the Laser



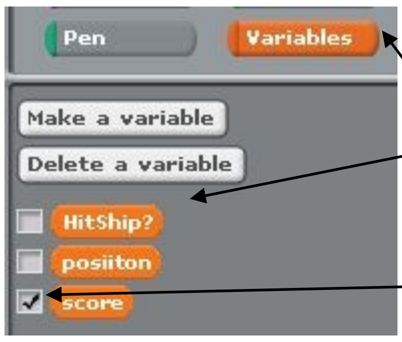
```
Laser
x: 21  y: 196  direction: 90
Scripts  Costumes  Sounds

when I receive fire
go to Ship
show
forever
    change y by 10
    if y position > 190
        hide
        stop script
```

With the laser we need to tell our program when to fire, where to position the laser when it is fired and how to move (later we will tell it what to do if it hits an enemy). Select the Laser sprite and then the *Scripts* tab for this Sprite. Select the **Control block** and build a new broadcast message called **fire.** Then build the blocks you see in the diagram that's to your left. Here is how this works. Later, we will attach a send fire message to the ships space bar key. When the space bar is pressed, the laser sprite gets the message and uses the **Motion Block** called go to Ship to center itself with the ship. It then use the (**Looks Block**) show to make sure it is visible on the screen. A forever (**Control block**) starts a loop where y is changed by 10 (added) each time through the loop making the laser beam go up the screen.

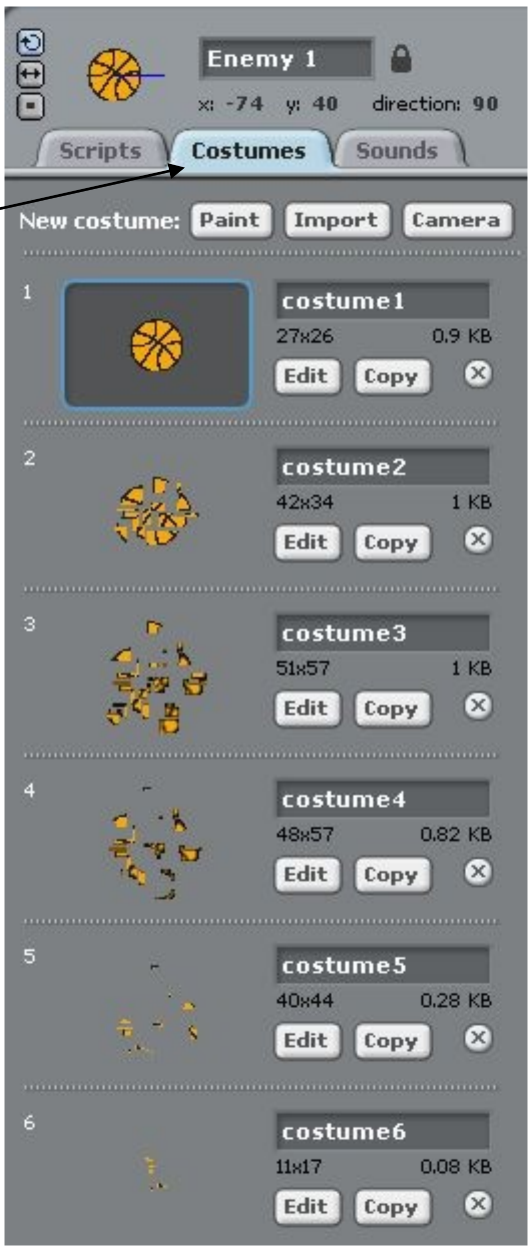
In the middle of the loop we check to see IF the lasers Y position is greater than 190 (the top of the screen).

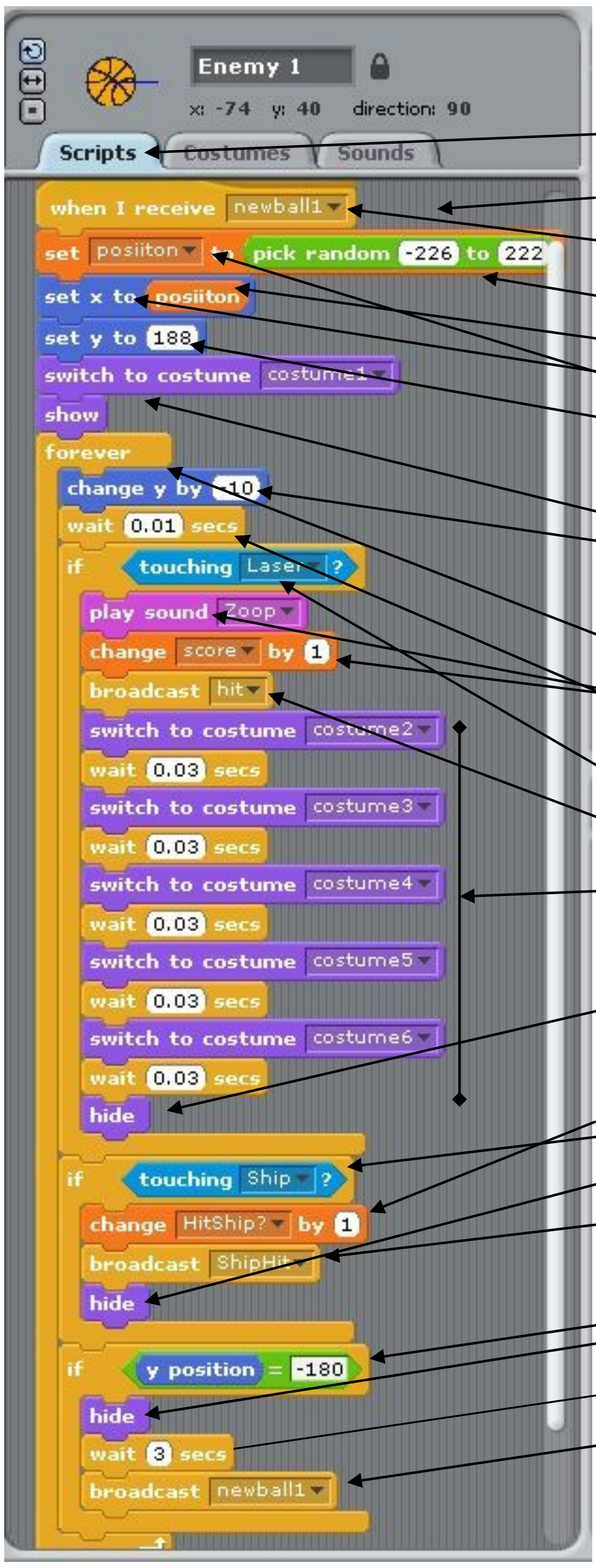
If it is, we hide the laser character (as it is now off the screen) and stop the script until it is called again by the user pressing the spacebar. When the user pressed the space bar again this whole process is repeated.

The Enemy Ship



Before writing the software for the enemy ship there are a few things we want to set-up. First, we want to create three **Variables** called "HitShip?, position and score". HitShip? will be used (later) to decide if our ship was hit by the enemy. Position will be used to randomly select a position for the enemy ship to fall from and score will be used to keep track of our games score. We will also "tick" the score variable so it appears on the stage to show our score. You can position the score on the stage. We will also create 6 *costumes* for our enemy player. Costume 1 is the regular costume for our player and costumes 2—6 are used to animate the enemy player being hit by the laser. If you want you can also skip this step and simply use a hide block with the enemy when it is hit. In my opinion the animation is nicer and is usually part of video games.


Enemy 1
x: -74 y: 40 direction: 90
Scripts
Costumes
Sounds
New costume: Paint Import Camera
1 costume1 27x26 0.9 KB Edit Copy
2 costume2 42x34 1 KB Edit Copy
3 costume3 51x57 1 KB Edit Copy
4 costume4 48x57 0.82 KB Edit Copy
5 costume5 40x44 0.28 KB Edit Copy
6 costume6 11x17 0.08 KB Edit Copy



```
when I receive newball1
set posiiton to pick random -226 to 222
set x to posiiton
set y to 188
switch to costume costume1
show
forever
    change y by -10
    wait 0.01 secs
    if touching Laser ?
        play sound Zoop
        change score by 1
        broadcast hit
        switch to costume costume2
        wait 0.03 secs
        switch to costume costume3
        wait 0.03 secs
        switch to costume costume4
        wait 0.03 secs
        switch to costume costume5
        wait 0.03 secs
        switch to costume costume6
        wait 0.03 secs
        hide
    if touching Ship ?
        change HitShip? by 1
        broadcast ShipHit
        hide
    if y position = -180
        hide
        wait 3 secs
        broadcast newball1
```

Writing the Software...

In the *Scripts* panel for Enemy 1 you are going to build the blocks you see in the diagram that's to your left. Here is how this works. Select the **Control block** and build a new broadcast message called **newball1.** When this broadcast is received (from the green flag) it will start a new enemy falling from the top. A random number is selected between –226 and 222 (the width of the top) and given to the position variable. The enemy's X (width) is then set to this and Y is set to 188 (so the ball is at the top out of sight). The costume is set to its regular costume1 and the enemy is shown on the screen.

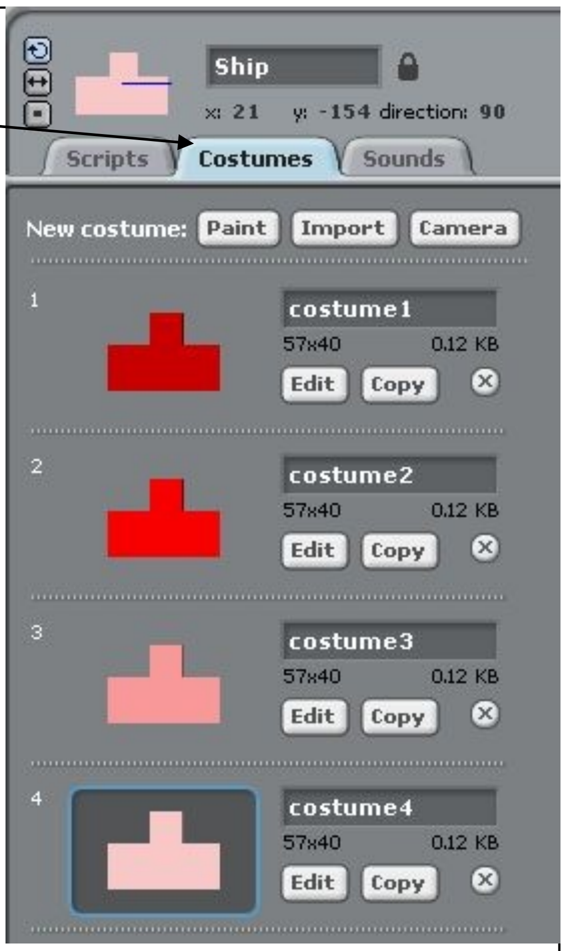
A forever loop starts to run and will move the enemy down the screen by 10 with a slight pause so it is not too fast. As the loop runs it will check IF certain things happen each time through the loop. The first things it checks is to see IF **the laser is touching the enemy**. If this is true then it plays a sound, adds 1 to the score, broad casts a message called hit (that makes the laser disappear) and goes through the costumes to make the ball look like it exploded and hides it at the end.

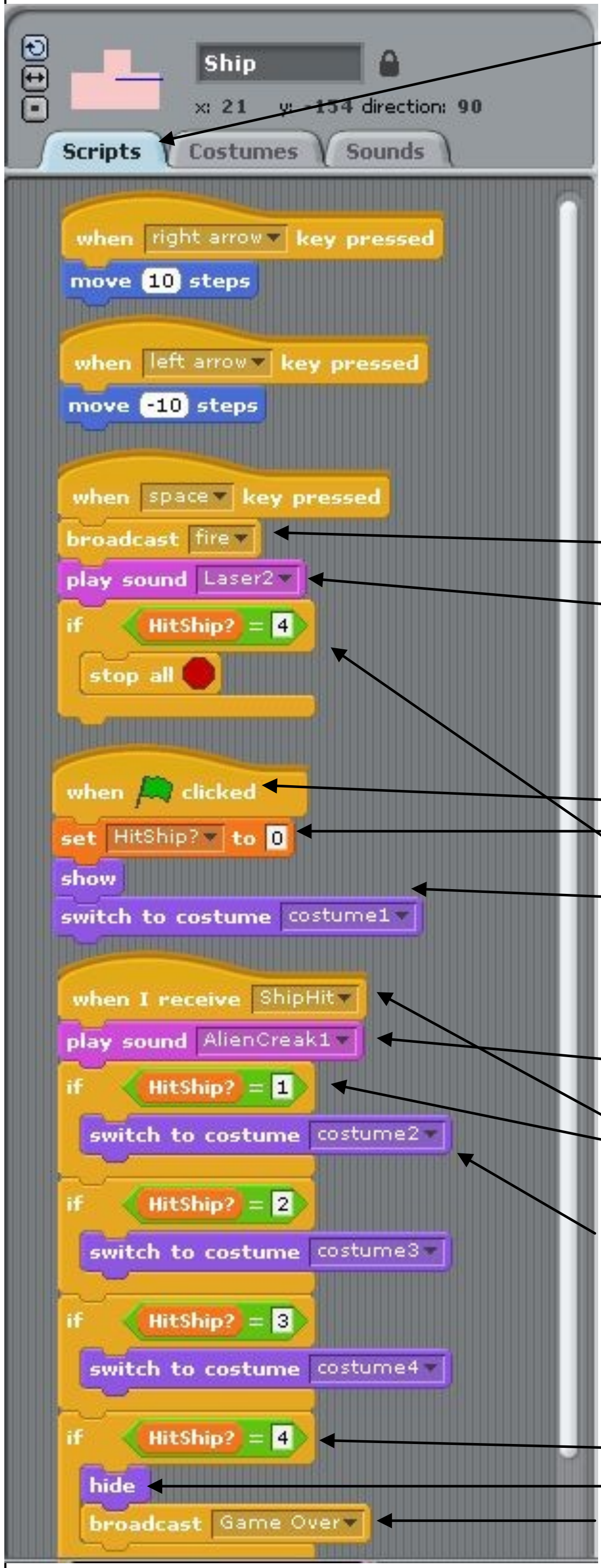
The loop then checks to see IF the enemy ship has hit our ship. If it has, then it adds 1 to the variable HitShip?(takes away one of our lives), broadcasts a ShipHit message and hides the enemy.

At the end of the loop if checks to see IF its Y position equals –180 (The bottom of the screen). If this is true it then hides the enemy ship waits for 3 seconds and broadcasts a message to itself (newball1) to tell it to start itself over again. Something giving itself an instruction to do something is a programming concept called "*recursion*" and is very common in video game development.

Finishing Our Ship

Before we finish our ship's software we will create 4 *costumes* for our ship. Costume 1 is the regular costume for our ship and costumes 2—4 are used show when our ship has been hit by an enemy player and how many lives are left. In the *Scripts* panel for Ship you are going to build the blocks you see in the diagram that's to your left.


Ship
x: 21 y: -154 direction: 90
Scripts
Costumes
Sounds
New costume: Paint Import Camera
1 costume1 57x40 0.12 KB Edit Copy
2 costume2 57x40 0.12 KB Edit Copy
3 costume3 57x40 0.12 KB Edit Copy
4 costume4 57x40 0.12 KB Edit Copy



```
when right arrow key pressed
move 10 steps

when left arrow key pressed
move -10 steps

when space key pressed
broadcast fire
play sound Laser2
if HitShip? = 4
    stop all

when ⚑ clicked
set HitShip? to 0
show
switch to costume costume1

when I receive ShipHit
play sound AlienCreak1
if HitShip? = 1
    switch to costume costume2
if HitShip? = 2
    switch to costume costume3
if HitShip? = 3
    switch to costume costume4
if HitShip? = 4
    hide
    broadcast Game Over
```

The arrow key were discussed earlier and there are no changes.

We will start by hooking up a fire broadcast to our space bar. This sends a message to the laser block and sets it going. We also added a sound once we have sent the fire broadcast (I'll cover the IF HitShip? = 4 later)

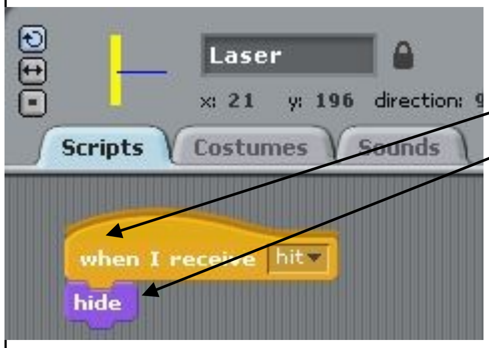
For the green flag block(Start Game) we will have it set HitShip? to 0 meaning none of our ships have been hit by the enemy (again I'll cover HitShip? later) show our ship, and makes sure its set to costume1 meaning our ship has not been hit yet.

Remembering back to our Enemy1, one of the things it did was if it ran into our ship (without being destroyed first) it sent a broadcast called ShipHit and added 1 to a variable called HitShip?. We will now use this with our Ship. Here is how it works. When a ShipHit message is received by out ship it plays a collision sound and checks the value of the HitShip? variable. Depending on the value of HitShip? It adjust the costume of our ship.

On the fourth time our ship has been hit (HitShip? = 4)it hides our ship and broadcasts a message called Game Over. Our ship's laser is disabled by the HitShip? = 4 block at the bottom of the Space Bar code that says **Stop All**.

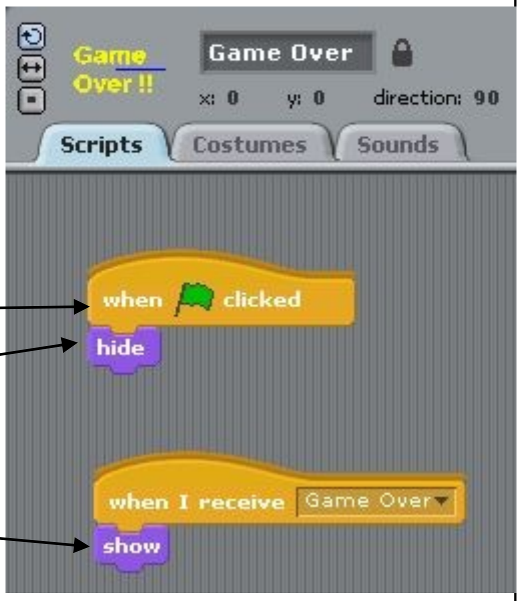
Finishing the Laser & Game Over

There are a couple of details we want to take care of before finishing our game. The first is in the Laser script. We want to add a block that allows it to receive a hit broadcast (from the enemy) and if it does, hide the laser. This is so when it hits an enemy it disappears and can fire again rather than continuing to the top of the screen.

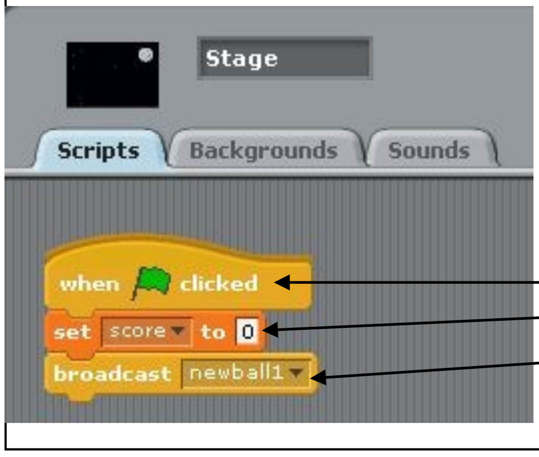


```
when I receive hit
hide
```

In the Game Over sprite we want to set it up so that when the flag is clicked (Start Game) the Game Over sign is hidden and when our ship gets more than 3 hits it places the game over on the screen (show).



```
when ⚑ clicked
hide

when I receive Game Over
show
```



```
when ⚑ clicked
set score to 0
broadcast newball1
```

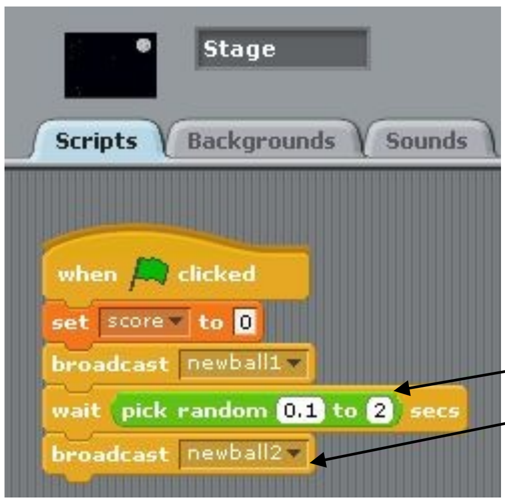
Finishing: Our Stage and Starting our Game

To finish our game select the stage and paint a background that you would like to use. Then select a flag control block and have it set the score (variable) to 0 when the game is started or restarted. It will then broadcast a newball1 message ***once*** to get the enemy ship(s) moving. Our enemy ship(s) will take care of this by itself (through recursion) for the remainder of the game.

Adding more Enemy Ships

Adding more enemy ships to your game is a simple process and involves 4 steps. First, select your enemy1 ship and duplicate it giving it a new name. Go to your new enemy ship and at the top, create a new broadcast message that it will receive. In my case I called this message newball2. Make this same change in the broadcast part of the loop (the recursion) at the bottom.

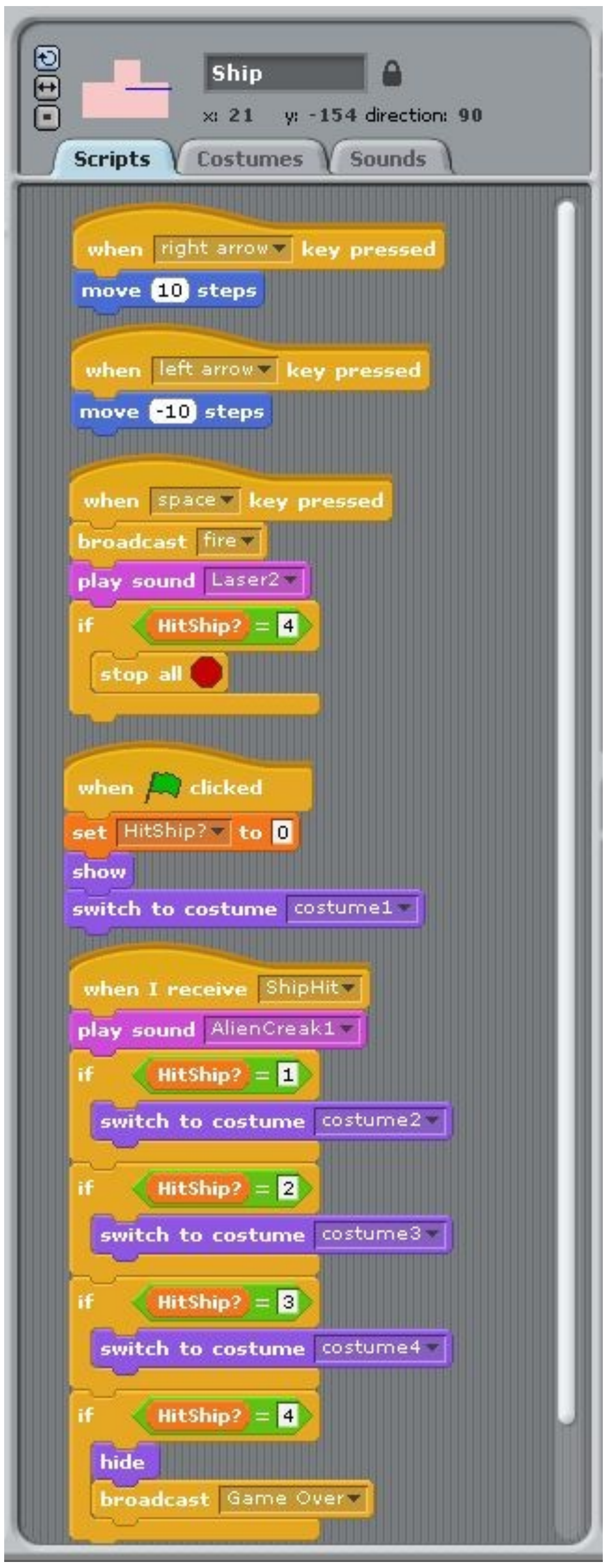
Then go to your stage. To have the ships fall at random intervals, have you program wait a random amount of time after your first ship starts then have it broadcast your newball2 (or whatever name you give it) to start your second enemy. You can add as many enemies as you would like this way.



```
when [flag] clicked
set score to 0
broadcast newball1
wait pick random 0.1 to 2 secs
broadcast newball2
```

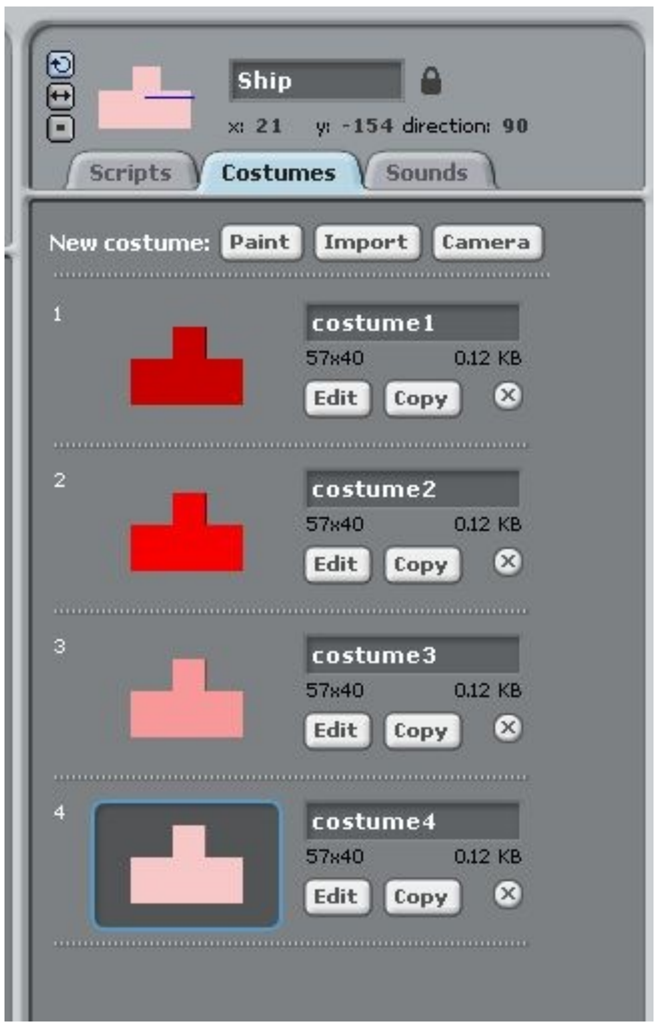
Complete Software Code

Main Ship Blocks



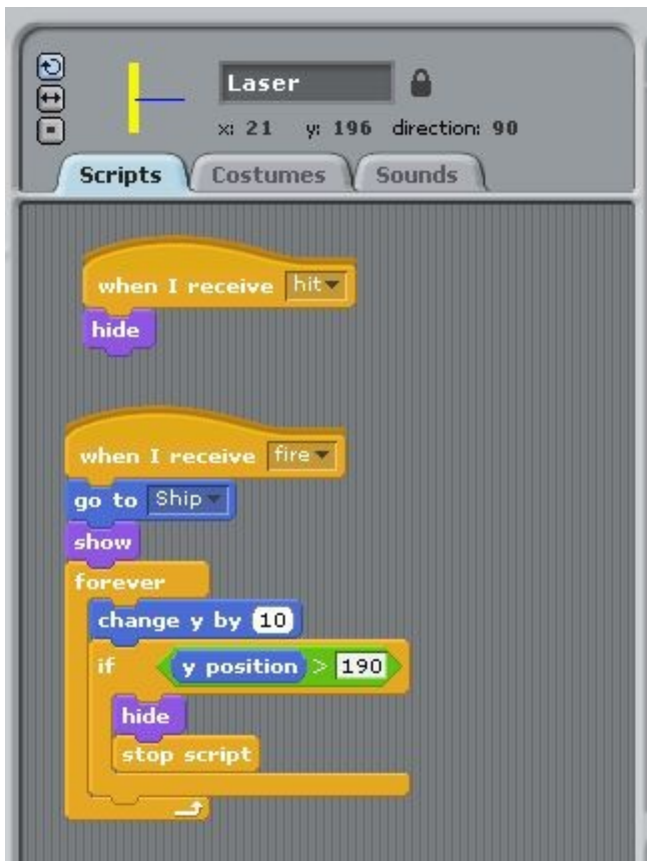
```
when right arrow key pressed
move 10 steps

when left arrow key pressed
move -10 steps

when space key pressed
broadcast fire
play sound Laser2
if HitShip? = 4
    stop all

when [flag] clicked
set HitShip? to 0
show
switch to costume costume1

when I receive ShipHit
play sound AlienCreak1
if HitShip? = 1
    switch to costume costume2
if HitShip? = 2
    switch to costume costume3
if HitShip? = 3
    switch to costume costume4
if HitShip? = 4
    hide
    broadcast Game Over
```

Main Ship Costumes



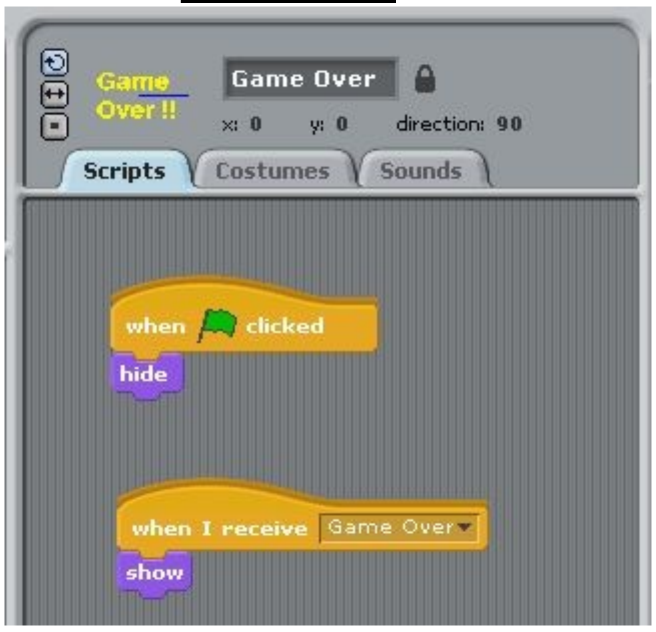
1 costume1 57x40 0.12 KB
2 costume2 57x40 0.12 KB
3 costume3 57x40 0.12 KB
4 costume4 57x40 0.12 KB

Laser Blocks



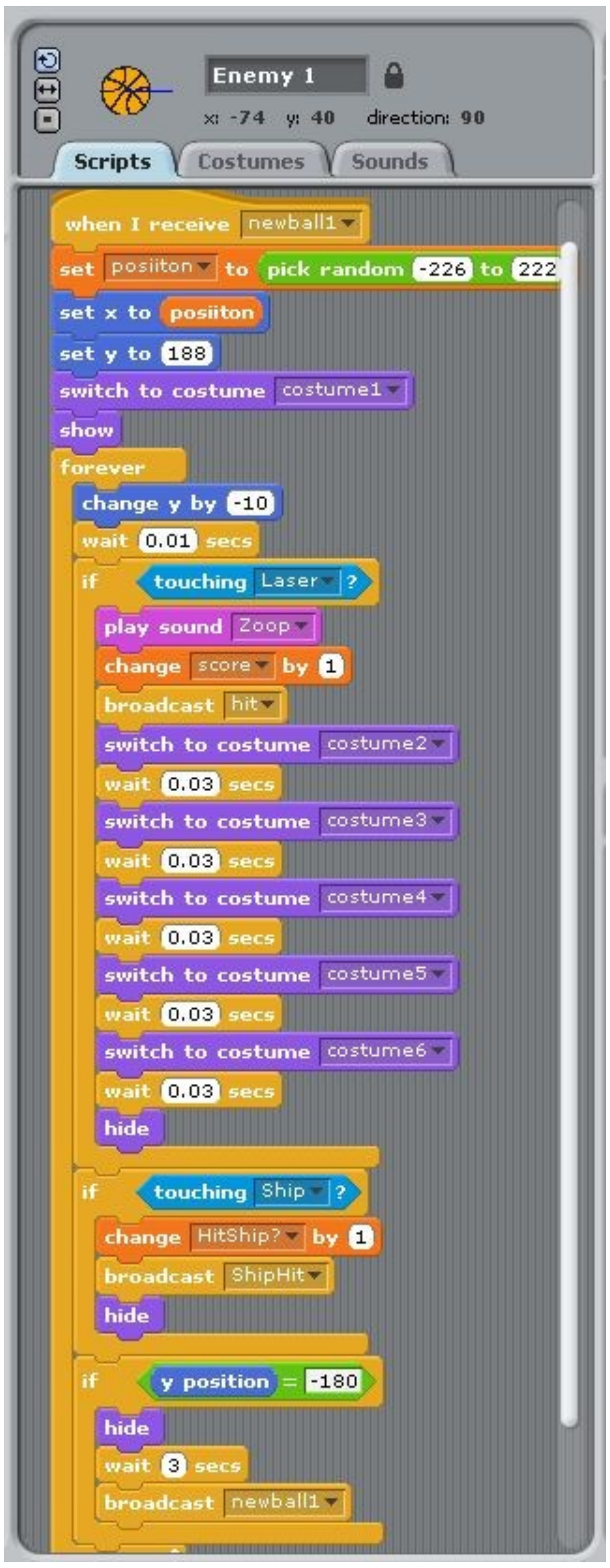
```
when I receive hit
hide

when I receive fire
go to Ship
show
forever
    change y by 10
    if y position > 190
        hide
        stop script
```

Game Over Blocks



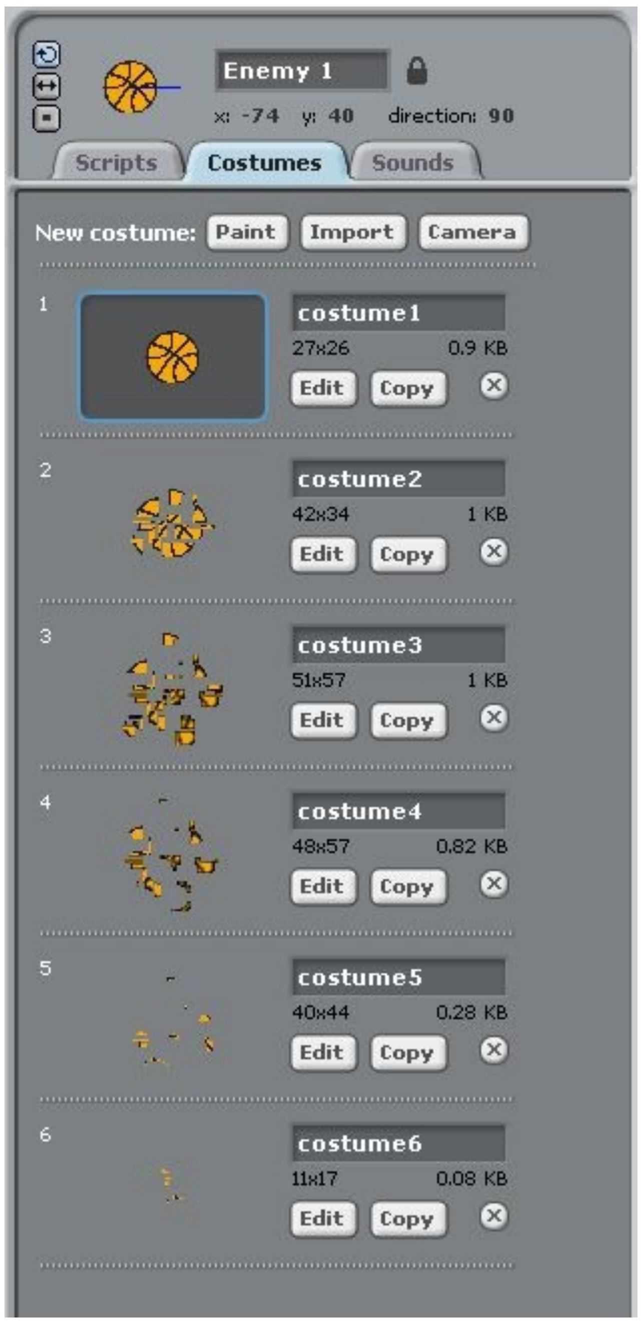
```
when [flag] clicked
hide

when I receive Game Over
show
```

Enemy Blocks



```
when I receive newball1
set posiiton to pick random -226 to 222
set x to posiiton
set y to 188
switch to costume costume1
show
forever
  change y by -10
  wait 0.01 secs
  if touching Laser ?
    play sound Zoop
    change score by 1
    broadcast hit
    switch to costume costume2
    wait 0.03 secs
    switch to costume costume3
    wait 0.03 secs
    switch to costume costume4
    wait 0.03 secs
    switch to costume costume5
    wait 0.03 secs
    switch to costume costume6
    wait 0.03 secs
    hide
  if touching Ship ?
    change HitShip? by 1
    broadcast ShipHit
    hide
  if y position = -180
    hide
    wait 3 secs
    broadcast newball1
```

Enemy Costumes



1 costume1 27x26 0.9 KB Edit Copy

2 costume2 42x34 1 KB Edit Copy

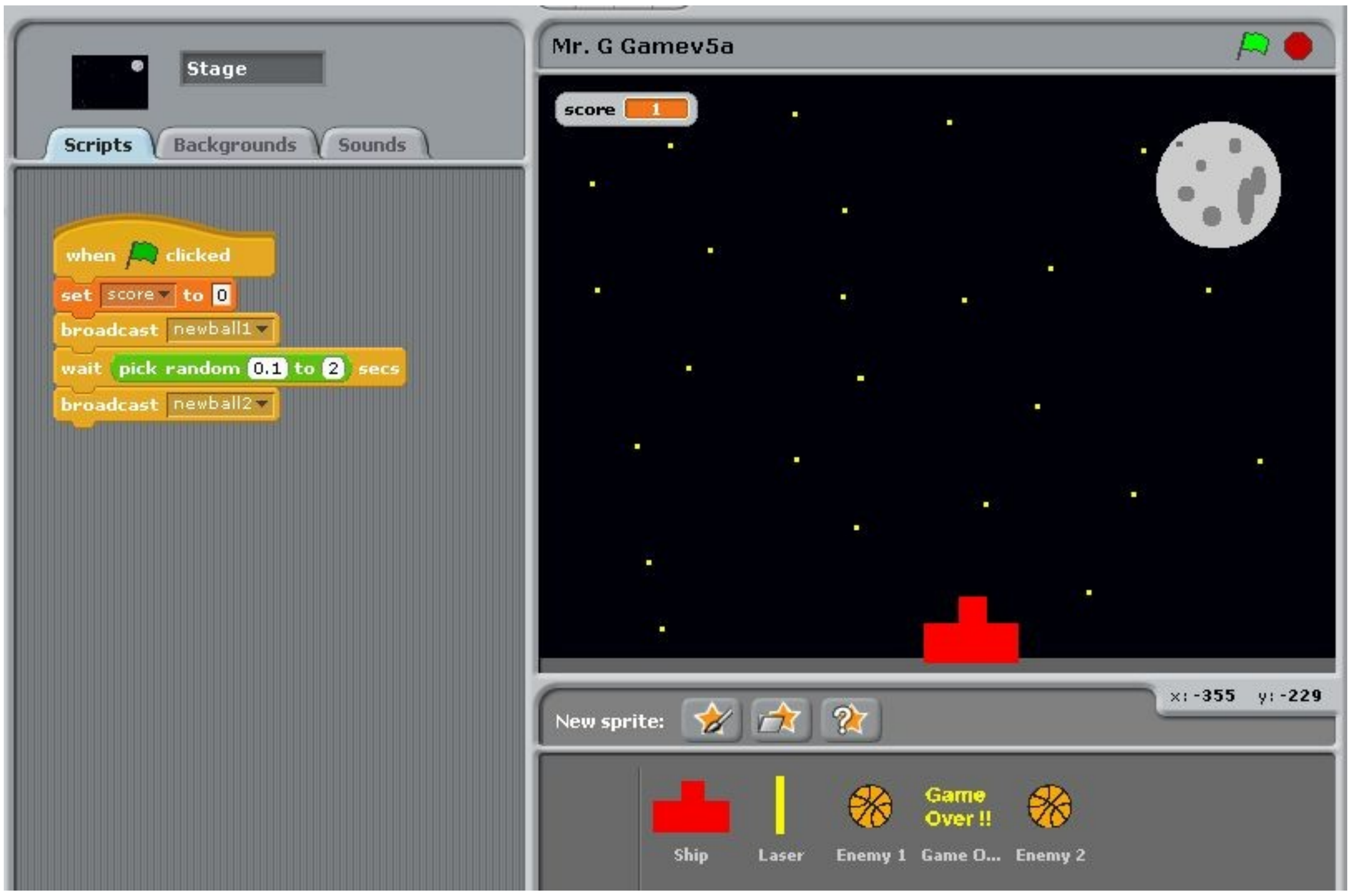
3 costume3 51x57 1 KB Edit Copy

4 costume4 48x57 0.82 KB Edit Copy

5 costume5 40x44 0.28 KB Edit Copy

6 costume6 11x17 0.08 KB Edit Copy

Stage Scripts & Sprites Blocks



```
when 🏁 clicked
set score to 0
broadcast newball1
wait pick random 0.1 to 2 secs
broadcast newball2
```