Public variables:

* Players starting X position
* Arrow keys and spacebar = false
* Enemy count = number
* Level = number
* Score = 0
* Setup = true
* Difficulty = empty
* Ship picture = newImage
* Game run = true
* shooting = false
* bullets = []
* enemies = []

Private Variables:

* self = this
* collided = false;
* enemy width = number
* enemy height = number
* enemy X position = number
* enemy Y position = number
* enemy Y speed = number
* delay = number
* delay rate = number
* bullet X position = number
* bullet Y position = number
* bulletYspeed = number

Constants:

* Left arrow key = 37
* Right arrow key = 39
* Space key = 32
* Player width = number
* Player height = number
* Player Y position = number
* Ship speed = number
* bullet width = number
* bullet height = number

function mainLoop {

if game is running {

colorRect game canvas

Draw player;

Run player movement;

check ship collision;

DrawText score

if setup is true and while difficulty is not set easy and difficulty is not set to hard(

prompt(choose difficulty, easy or hard.)

if number of enemies is more than 0 run draw enemy function and enemy movefunction

if the number of bullets is greater than 0 run draw bullet and bullet move function

if (bullet is out of bounds or it has collided)

(then delete the bullet)

(Clear the bullets from the array instead of leaving them as a placeholder)

If the game isn’t running then:

colorRect gamecanvas to black

drawText – ‘your score was’ + score public variable

End of main loop function)

**class Player** (

constructor(image source, player X position, player Y position, playerwidth, playerheight, playerspeed)

this.source = source;

this.x = x;

this.y = y;

this.w = w;

this.h = h;

this.playerspeed = playerspeed;

}

drawShip(

canvascontext.drawImage(this.source, this.x, this.y, this.w, this.h);

)

Player movement(

if the left key is pressed (

player x - X speed

)

if the right key is pressed (

player x + X speed

)

enemyHit(item) (

return (this.x less than or equal to (enemy.x + enemy.width) and

(this.x + this.width) is greater than or equal to enemy.x) and

(this.y less than or equal to (enemy.y + enemy.height) and

(this.y + this.height) is greater than or equal to enemy.y)

)

Has collided(

var self = this;

var collided = false;

enemies.forEach(function (enemy, i) {

if (self.hasHitEnemy(enemy)) {

if (self.x + self.w greater than enemy.x and self.x less than enemy.x + enemy.w and self.y + self.h greater than enemy.y and self.y less than enemy.y + enemy.h) {

gameRun = false;

function keypressed event (

if key code = left key

left key pressed = true

)

function keypressed event (

if key code = right key

right key pressed = true

)

function keypressed event (

if key code = space key

space key pressed = true

)

function keyreleased event (

if key code = left key

left key pressed = false

)

function keyreleased event (

if key code = right key

right key pressed = false

)

function keyreleased event (

if key code = space key

space key pressed = false

run draw bullets function

)

End of player class

**class Enemy**:

(constructor(X position, Y position, width, height, colour, Y speed, delay, delay rate)

this.x = x;

this.y = y;

this.w = w;

this.h = h;

this.c = c;

this.ySpeed = ySpeed;

this.delay = delay;

this.delayRate = delayRate;

)

Draw enemy (

Canvas context.fillstyle = this.colour;

canvascontext.fillRect(this.x, this.y, this.w, this.h)

)

Enemy movement(

this.y + this.ySpeed

if (this.y is greater than the canvas height) {

this.y = 0

this.x = Math.floor(Math.random() \* canvas width - this.w)

this.ySpeed = Math.floor(Math.random() \* (max - min) + min)

End of enemy class

function draw enemies(

var enemy width = number

var enemy height = number

var enemy X position = Math.floor(Math.random() \* canvaswidth – enemy width

var enemy Y position = 0 – enemy height

var enemy Y speed = Math.floor(Math.random() \* (max - min) + min)

var e = new Enemy(Enemy X position, Enemy Y positon, enemy width, enemyheight, colour, enemy Y speed)

enemies (array) .push(e)

)

if (difficulty = easy)

enemycount = 5

}

if (difficulty = hard)

enemycount = 9

}

**Class bullet:**

constructor(x, y, width, height, colour, Y speed)

this.x = x;

this.y = y;

this.w = w;

this.h = h;

this.c = c;

this.ySpeed = ySpeed;

)

drawBullet func (

canvas context.fillStyle = this.c

canvas context.fillRect (this.x, this.y, this.w, this.h)

bullet movement func (

this.y - this.ySpeed

)

Out of bounds func (

return this.y is less than 0 or this.y greater than canvas height or this.x less than 0 or this.x greater than canvas width;

)

Has hit item func (

return this.x + this.width) greater than or equal to item.x and this.x less than or equal to (item.x + item.w))

and

(this.y greater than item.y and this.y less than or equal to (item.y + item.h)

)

Has hit enemy (enemy) (

return this has hit Item (enemy)

}

Has collided func (

var self = this;

var collided = false;

enemies.forEach(func (enemy, i) {

if (self.has hit enemy(enemy)

delete enemies[i]

collided = true

score + 1

run draw enemies function

Clear the bullets from the array instead of leaving them as a placeholder