window.onload = function() {

//Create a new Phaser game, with dimentions of 800px wide and 600px tall

var game= new Phaser.Game(800, 600, Phaser.AUTO, 'game-world', {preload: preload, create: create, update: update });

var gameState = 0 //game is starting

var bg1c, bg2c;

var bg1bw, bg2bw;

var gameState = 0; //Game is starting

var level = 1; //Current game level

var button; //The menu button

var environmentGroup; //Group for all world objects such as platforms and background

var playerGroup; //Group for the player and and player info elements

var bg1

var platforms; //Group of platforms

var MenuPGroup;

//Player gobal variables

var player; //The player object

var speed = 10; //The speed the player moves at

var MAX\_SPEED = 300; //The maximum speed of the player

var jumpForce = 350; //The force that the player will jump with

var controls; //The controls for the player

var isPaused = false; //pause

//Preload function, where we can load all the assets that will be used

function preload(){

game.load.spritesheet('player', 'assets/Character.png', 96, 128); //Player sprite

game.load.spritesheet('MenuP', 'assets/Character.png', 96, 128);

game.load.spritesheet('player', 'assets/Character.png', 25, 36); //Player sprite

game.load.spritesheet('player', 'assets/Character.png', 25, 36); //Player sprite

game.load.image('background1', 'assets/bg1.2colour.png'); //background with colour

game.load.image('background2', 'assets/bg1.2bw.png'); //load black and white background

game.load.image('button', 'assets/button1.png'); //Start button

game.load.image('menu', 'assets/StartScreen2.png'); //Menu background

game.load.image('rock', 'assets/rock.png'); //rock

game.load.image('ground','assets/dirtground1.png'); //dirt (brown)

game.load.image('missile', 'assets/missile.png');

} //END of preload

//Create function, where all the initial objects are created

function create(){

game.physics.startSystem(Phaser.Physics.ARCADE);

//create groups

environmentGroup = game.add.group();

playerGroup = game.add.group();

platforms = game.add.group();

platforms.enableBody = true;

//load the menu

loadMenu();

//Enable the Arcade physics system

game.physics.startSystem(Phaser.Physics.ARCADE);

controls = game.input.keyboard.addKeys(

{

'jump': Phaser.KeyCode.W,

'left': Phaser.KeyCode.A,

'right': Phaser.KeyCode.D,

'crouch': Phaser.KeyCode.S,

'pause': Phaser.KeyCode.SPACEBAR

}

);

} //END of create

//Update function, runs every frame

function update(){

if(gameState == 0) { //Game menu code

//Animate the player moving across the bottom of the screen

MenuP.x++; //Move the player right

if (MenuP.x > game.world.width) { //If the player has moved off the right edge of the screen

MenuP.x = -48; //Place the player just to the left of the screen

}

} else if(gameState == 1) { //Game code

if(controls.left.isDown){

//player.body.velocity.x = -150;

//player.animations.play('left');

} else if(controls.right.isDown){

//player.body.velocity.x = 150;

//player.animations.play('right');

} else {

//player.animations.stop();

//player.frame = 3;

//player.body.velocity.x = 0;

}

if(controls.jump.isDown && player.body.touching.down && hittingPlatform) {

player.body.velocity.y = -350;

}

//If the SPACEBAR is pressed down (not held down)

if(controls.pause.justDown){

isPaused = !isPaused; //False becomes true, and True becomes false

game.physics.arcade.isPaused = isPaused; //Disables/Enables physics events

console.log("Game paused: " + isPaused);

}

//If the game is paused, skip the rest of the update() function

if(isPaused){

return;

}

if(bg1c.x < -game.width) {

bg1c.x = game.width;

} else if (bg2c.x < -game.width){

bg2c.x = game.width;

}

}

} //end of update

function loadMenu(){

bg = game.add.sprite(0,0,'menu');

environmentGroup.add(bg); //Add the bg to the environmentGroup

//Add a play button

button = game.add.button(game.world.centerX, game.world.centerY, 'button');

button.anchor.setTo(0.45,-0.600);

button.onInputUp.add(actionPlay); //When the button is released

environmentGroup.add(button);

//Place some text on top of the button

var text = game.add.text(button.x,button.y);

text.anchor.setTo(0.5,0.5);

environmentGroup.add(text);

//note this is a test sprite (Do not remove or use)

//temp = game.add.sprite(50,50,'player');

//temp.animations.add('left', [1,2,3,4,5,6,7], 5, true);

//temp.animations.play('left');

//Do some initial player set up

MenuP = game.add.sprite(-48, game.world.height-280, 'player');

//player.body.collideWorldBounds = true;

//Add animation sequences to the player object

MenuP.animations.add('right', [4,5,6,7], 5, true);

MenuP.animations.play('right'); //Start playing the 'right' animation

MenuPGroup = game.add.group();

MenuPGroup.add(MenuP); //Add the player to the player group

} //end of loadMenu

function loadLevelOne(){

gameState = 1;

bg1c = game.add.sprite(0,0,'background2')

bg2c = game.add.sprite(game.width,0,'background2');

environmentGroup.add(bg1c);

environmentGroup.add(bg2c);

var ledge = platforms.create(0, 448, 'ground');

ledge.body.immovable = true;

game.physics.arcade.enable(bg1c);

bg1c.body.velocity.x = -20;

game.physics.arcade.enable(bg2c);

bg2c.body.velocity.x = -20;

player = game.add.sprite(0, game.world.height-280, 'player');

player.x = 50;

game.physics.enable(player);

playerGroup.add(player);

player.animations.add('left', [0,1,2,3], 8, true);

player.animations.add('right', [4,5,6,7], 8, true);

player.animations.play('right');

weapon = game.add.weapon(5, 'missile');

weapon.bulletKillType = Phaser.Weapon.KILL\_WORLD\_BOUNDS;

weapon.bulletScale = 0.1; //Scale size of bullets

weapon.fireRate = 500; //Milliseconds between shots

weapon.trackSprite(player);

if (game.input.activePointer.leftButton.isDown) {

weapon.fireAtPointer();

}

}

function unloadLevel(){

environmentGroup.removeAll(true);

MenuPGroup.removeAll(true);

}

function actionPlay(){

unloadLevel();

loadLevelOne();

}

}; //end of program