```actionscript
package{

import flash.display.MovieClip;

import flash.events.KeyboardEvent;

import flash.ui.Keyboard;

import flash.events.Event; //used for ENTER_FRAME event


public class Main extends MovieClip{


//constants

const gravity:Number = 1.5;            //gravity of the game

const dist_btw_obstacles:Number = 300; //distance between two obstacles

const ob_speed:Number = 8;             //speed of the obstacle

const jump_force:Number = 15;          //force with which it jumps


//variables

var player:Player = new Player();

var lastob:Obstacle = new Obstacle();  //varible to store the last obstacle in the obstacle array

var obstacles:Array = new Array();     //an array to store all the obstacles

var yspeed:Number = 0;                 //A variable representing the vertical speed of the bird

var score:Number = 0;                  //A variable representing the score


public function Main(){

init();

}


function init():void {

//initialize all the variables

player = new Player();

lastob = new Obstacle();

obstacles = new Array();

yspeed = 0;

score = 0;


//add player to center of the stage the stage

player.x = stage.stageWidth/2;

player.y = stage.stageHeight/2;

addChild(player);
```

```actionscript
//create 3 obstacles ()
createObstacle();
createObstacle();
createObstacle();

//Add EnterFrame EventListeners (which is called every frame) and KeyBoard EventListeners
addEventListener(Event.ENTER_FRAME,onEnterFrameHandler);
stage.addEventListener(KeyboardEvent.KEY_UP, key_up);
}

private function key_up(event:KeyboardEvent){
if(event.keyCode == Keyboard.SPACE){
//If space is pressed then make the bird
yspeed = -jump_force;
}
}

function restart(){
if(contains(player))
removeChild(player);
for(var i:int = 0; i < obstacles.length; ++i){
if(contains(obstacles[i]) && obstacles[i] != null)
removeChild(obstacles[i]);
obstacles[i] = null;
}
obstacles.slice(0);
init();
}

function onEnterFrameHandler(event:Event){
//update player
yspeed += gravity;
player.y += yspeed;

//restart if the player touches the ground
if(player.y + player.height/2 > stage.stageHeight){
restart();
}

//Don't allow the bird to go above the screen
```

```actionscript
if(player.y - player.height/2 < 0){

player.y = player.height/2;

}


//update obstacles
for(var i:int = 0;i<obstacles.length;++i){

updateObstacle(i);

}


//display the score
scoretxt.text = String(score);

}


//This functions update the obstacle
function updateObstacle(i:int){

var ob:Obstacle = obstacles[i];


if(ob == null)

return;

ob.x -= ob_speed;


if(ob.x < -ob.width){

//if an obstacle reaches left of the stage then change its position to the back of the last obstacle

changeObstacle(ob);

}


//If the bird hits an obstacle then restart the game
if(ob.hitTestPoint(player.x + player.width/2,player.y + player.height/2,true)

|| ob.hitTestPoint(player.x + player.width/2,player.y - player.height/2,true)

|| ob.hitTestPoint(player.x - player.width/2,player.y + player.height/2,true)

|| ob.hitTestPoint(player.x - player.width/2,player.y - player.height/2,true)){

restart();

}


//If the bird got through the obstacle without hitting it then increase the score
if((player.x - player.width/2 > ob.x + ob.width/2) && !ob.covered){

++score;

ob.covered = true;

}

}
```

```actionscript
//This function changes the position of the obstacle such that it will be the last obstacle and it also
randomizes its y position
function changeObstacle(ob:Obstacle){
ob.x = lastob.x + dist_btw_obstacles;
ob.y = 100+Math.random()*(stage.stageHeight-200);
lastob = ob;
ob.covered = false;
}


//this function creates an obstacle
function createObstacle(){
var ob:Obstacle = new Obstacle();
if(lastob.x == 0)
ob.x = 800;
else
ob.x = lastob.x + dist_btw_obstacles;
ob.y = 100+Math.random()*(stage.stageHeight-200);
addChild(ob);
obstacles.push(ob);
lastob = ob;
}



}
}
```

RAW Paste Data

```actionscript
package{

 import flash.display.MovieClip;

 import flash.events.KeyboardEvent;

 import flash.ui.Keyboard;

 import flash.events.Event; //used for ENTER_FRAME event


 public class Main extends MovieClip{
```

```
//constants
```