using System;

using System.Collections.Generic;

using System.Drawing;

using System.Windows.Forms;

namespace PlaneGameC\_

{

public partial class Form1 : Form

{

private int \_playerSpeed = 5;

private int \_enemySpeed = 3;

private int \_bulletSpeed = 5 ;

private int \_score = 0;

private bool \_moveLeft = false;

private bool \_moveRight = false;

private List<PictureBox> \_bullets = new List<PictureBox>();

public Form1()

{

InitializeComponent();

InitGame();

}

private void InitGame()

{

gameTimer.Interval = 20;

gameTimer.Tick += gameTimer\_Tick;

gameTimer.Start();

}

private void Form1\_Load(object sender, EventArgs e)

{

}

private void gameTimer\_Tick(object sender, EventArgs e)

{

if (\_moveLeft && player.Left > 0)

player.Left -= \_playerSpeed;

if (\_moveRight && player.Right < this.ClientSize.Width)

player.Left += \_playerSpeed;

obstacle1.Top += \_enemySpeed;

obstacle2.Top += \_enemySpeed;

if (obstacle1.Top > this.ClientSize.Height)

ResetEnemy(obstacle1);

if (obstacle2.Top > this.ClientSize.Height)

ResetEnemy(obstacle2);

if (player.Bounds.IntersectsWith(obstacle1.Bounds) || player.Bounds.IntersectsWith(obstacle2.Bounds))

{

GameOver();

return;

}

for (int i = \_bullets.Count - 1; i >= 0; i--)

{

var bullet = \_bullets[i];

bullet.Top -= \_bulletSpeed;

if (bullet.Top < 0)

{

this.Controls.Remove(bullet);

\_bullets.RemoveAt(i);

continue;

}

if (bullet.Bounds.IntersectsWith(obstacle1.Bounds))

{

DestroyEnemy(obstacle1);

this.Controls.Remove(bullet);

\_bullets.RemoveAt(i);

}

else if (bullet.Bounds.IntersectsWith(obstacle2.Bounds))

{

DestroyEnemy(obstacle2);

this.Controls.Remove(bullet);

\_bullets.RemoveAt(i);

}

}

scoreLabel.Text = "Score: " + \_score;

}

private void GameOver()

{

gameTimer.Stop();

MessageBox.Show("Game Over! Your score: " + \_score, "Game Over");

Application.Exit();

}

private void DestroyEnemy(PictureBox enemy)

{

ResetEnemy(enemy);

\_score += 10;

}

private void ResetEnemy(PictureBox enemy)

{

Random rand = new Random();

enemy.Top = 0;

enemy.Left = rand.Next(0, this.ClientSize.Width - enemy.Width);

}

private void FireBullet()

{

PictureBox bullet = new PictureBox

{

Size = new Size(10, 5),

BackColor = Color.Black,

Left = player.Left + (player.Width / 2) - 5,

Top = player.Top - 10

};

\_bullets.Add(bullet);

this.Controls.Add(bullet);

}

private void Form1\_KeyDown(object sender, KeyEventArgs e)

{

if (e.KeyCode == Keys.A || e.KeyCode == Keys.Left)

\_moveLeft = true;

if (e.KeyCode == Keys.D || e.KeyCode == Keys.Right)

\_moveRight = true;

if (e.KeyCode == Keys.Space)

FireBullet();

}

private void Form1\_KeyUp(object sender, KeyEventArgs e)

{

if (e.KeyCode == Keys.A || e.KeyCode == Keys.Left)

\_moveLeft = false;

if (e.KeyCode == Keys.D || e.KeyCode == Keys.Right)

\_moveRight = false;

}

}

}