Code indien er iets mis zou zijn met het project

Mainwindow

public partial class MainWindow : Window

{

//declaratie

DispatcherTimer time;

//lijst items van de slang en het voedsel

List<Snake> snakebody;

List<Food> food;

Random rd = new Random();

double x = 100;

double y = 100;

public int score = 0;

int direction = 0;

int count = 0;

bool EindschermKanOpenen = false;

//waardes van de bewegingen

int right = 2;

int left = 4;

int up = 6;

int down = 8;

//de beginwaardes worden gegeven

public MainWindow()

{

InitializeComponent();

time = new DispatcherTimer();

snakebody = new List<Snake>();

food = new List<Food>();

snakebody.Add(new Snake(x, y));

food.Add(new Food(rd.Next(0, 37) \* 10, rd.Next(0, 35) \* 10));

//De snelheid van de slang

time.Interval = new TimeSpan(0, 0, 0, 0, 55);

time.Tick += time\_Tick;

}

//Voegt "voedsel" toe in het venster

void VoegVoedselToe()

{

food[0].PlaatsVoedsel();

mycanvas.Children.Insert(0, food[0].ellipse);

}

//zet het 1ste bolletje van de slang in het venster

void VoegSlangToe()

{

foreach (Snake snake in snakebody)

{

snake.PlaatsSlang();

mycanvas.Children.Add(snake.ellipse);

}

}

//deze acties worden uitgevoerd bij elke tick van de timer

void time\_Tick(object sender, EventArgs e)

{

//als de speler niets indrukt gaat de slang niet van richting veranderen

if (direction != 0)

{

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

{

snakebody[i] = snakebody[i - 1];

}

}

//(int)Richting roept een waarde in het EnumSlangRichting script aan

if (direction == up)

y -= 10;

if (direction == down)

y += 10;

if (direction == left)

x -= 10;

if (direction == right)

x += 10;

//slang eet voedsel, er komt nieuw voedsel op het spelvenster en het oude voedsel gaat weg

//score wordt opgeteld en in het txt vakje van de score gezet

if (snakebody[0].x == food[0].x && snakebody[0].y == food[0].y)

{

snakebody.Add(new Snake(food[0].x, food[0].y));

food[0] = new Food(rd.Next(0, 33) \* 10, rd.Next(0, 30) \* 10);

mycanvas.Children.RemoveAt(0);

VoegVoedselToe();

score++;

txtbScore.Text = score.ToString();

}

snakebody[0] = new Snake(x, y);

//raakt de slang de rand?

if (!EindschermKanOpenen && (snakebody[0].x > 370 || snakebody[0].y > 350 || snakebody[0].x < 0 || snakebody[0].y < 0))

{

EindschermKanOpenen = true;

VulNaamIn vulnaamin = new VulNaamIn();

this.Hide();

vulnaamin.Show();

}

//raakt de slang zijn eigen lichaam?

for (int i = 1; i < snakebody.Count; i++)

{

if (!EindschermKanOpenen && (snakebody[0].x == snakebody[i].x && snakebody[0].y == snakebody[i].y))

{

EindschermKanOpenen = true;

VulNaamIn vulnaamin = new VulNaamIn();

this.Hide();

vulnaamin.Show();

}

}

//toevoegen van een bolletje

for (int i = 0; i < mycanvas.Children.Count; i++)

{

if (mycanvas.Children[i] is Ellipse)

count++;

}

mycanvas.Children.RemoveRange(1, count);

count = 0;

VoegSlangToe();

}

//detecteerd wanneer er een knop op het toetsenbord ingedrukt wordt

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

{

//als de ingeduwde knop het pijltje naar boven is wordt er gekeken of de slang niet naar beneden gaat

//indien deze niet naar beneden gaat zal de slang naar boven gaan

//dit is zo omdat de slang geen contact mag maken met zijn lichaam, en anders door zichzelf heen zou gaan

if (e.Key == Key.Up && direction != down)

direction = up;

if (e.Key == Key.Down && direction != up)

direction = down;

if (e.Key == Key.Left && direction != right)

direction = left;

if (e.Key == Key.Right && direction != left)

direction = right;

}

//message box met informatie als de game begint

private void Window\_Loaded(object sender, RoutedEventArgs e)

{

MessageBox.Show("Beweeg de slang doorheen het veld, en vang het voedsel" + "\n" +

"Als je de randen, of het lichaam van de slang zelf raakt, heb je gefaald" + "\n" + "\n" +

"Gebruik de pijltjestoetsen om je doorheen het veld te bewegen" + "\n" + "\n" +

"Om te starten:" + "\n" + "Druk op start en gebruik één van de pijltjestoetsen ");

}

//als er op start gedrukt wordt verschijnt de slang en het voedsel en kan je beginnen

private void BtnStart\_Click(object sender, RoutedEventArgs e)

{

VoegSlangToe();

VoegVoedselToe();

time.Start();

}

//opent het highscore menu

private void BtnHighscores\_Click(object sender, RoutedEventArgs e)

{

Highscores highscores = new Highscores();

this.Close();

highscores.Show();

}

//beeindigd de game

private void BtnExit\_Click(object sender, RoutedEventArgs e)

{

this.Close();

}

}

Klasse: Food

namespace SnakeGame

{

public class Food

{

public double x, y;

public Ellipse ellipse = new Ellipse();

//constructor

public Food(double x, double y)

{

this.x = x;

this.y = y;

}

public double X

{

get { return x; }

}

public double Y

{

get { return y; }

}

//deze functie geeft het voedsel een breedte, groote, kleur en een plaats

public void PlaatsVoedsel()

{

ellipse.Width = ellipse.Height = 10;

ellipse.Fill = Brushes.Red;

//geeft linker en top waarde van het object

Canvas.SetLeft(ellipse, x);

Canvas.SetTop(ellipse, y);

}

}

}

Klasse: Snake

namespace SnakeGame

{

public class Snake

{

public double x, y;

public Ellipse ellipse = new Ellipse();

//constructor

public Snake(double x, double y)

{

this.x = x;

this.y = y;

}

public double X

{

get { return x; }

}

public double Y

{

get { return Y; }

}

//deze fucntie geeft de slang een grootte, breedte, kleur en een plaats

public void PlaatsSlang()

{

ellipse.Width = ellipse.Height = 12;

ellipse.Fill = Brushes.White;

//geeft linker en top waarde van het object

Canvas.SetLeft(ellipse, x);

Canvas.SetTop(ellipse, y);

}

}

}

Highscores Window

public partial class Highscores : Window

{

//de database wordt aangeroepen

SqlConnection scnnHighscores = new SqlConnection(Properties.Settings.Default.SnakeConnection);

SqlCommand scmdtop10 = new SqlCommand();

//datareader overloopt alle gegevens in de database

SqlDataReader sdrHighscores;

//De data van de vorige spelers worden in het highscore venster geladen

public Highscores()

{

InitializeComponent();

scmdtop10.Connection = scnnHighscores;

scnnHighscores.Open();

scmdtop10.CommandText = "SELECT TOP 10 \* FROM tblScore ORDER BY Score DESC";

sdrHighscores = scmdtop10.ExecuteReader();

//elk gelezen gegeven (in de top 10) wordt toegevoegd aan de list

while (sdrHighscores.Read())

{

string strlijn;

strlijn = "Speler " + sdrHighscores["Naam"].ToString() + ": \t\t" + sdrHighscores["Score"].ToString() + " punten";

lstHighscores.Items.Add(strlijn);

}

//connectie sluiten

scnnHighscores.Close();

}

//de game wordt afgesloten

private void BtnExit\_Click(object sender, RoutedEventArgs e)

{

this.Close();

}

//de game start opnieuw

private void BtnOpnieuw\_Click(object sender, RoutedEventArgs e)

{

MainWindow main = new MainWindow();

main.score = 0;

main.txtbScore.Text = "0";

this.Hide();

main.Show();

}

}

VulnaamInWindow

public partial class VulNaamIn : Window

{

//de database wordt aangeroepen

SqlConnection snnScore = new SqlConnection(Properties.Settings.Default.SnakeConnection);

SqlCommand cmd = new SqlCommand();

//het mainwindow wordt opgeroepen om de score door te geven aan de DB

MainWindow MainWindow = (MainWindow)Application.Current.MainWindow;

public VulNaamIn()

{

InitializeComponent();

}

//de naam en de score van de speler worden opgeslagen

private void BtnOpslaan\_Click(object sender, RoutedEventArgs e)

{

snnScore.Open();

cmd.Connection = snnScore;

cmd.CommandText = "INSERT INTO tblScore(Naam, Score) VALUES('" + txtNaam.Text + "','" + MainWindow.txtbScore.Text + "')";

cmd.ExecuteNonQuery();

snnScore.Close();

MessageBox.Show("Uw score is opgeslagen!");

}

//als er op de highscores knop gedrukt wordt opent het highscores window

//dit window sluit

private void BtnHighscores\_Click(object sender, RoutedEventArgs e)

{

Highscores highscores = new Highscores();

this.Close();

highscores.Show();

}

//de game wordt gesloten

private void BtnExit\_Click(object sender, RoutedEventArgs e)

{

this.Close();

}

}