

XNA – MY FIRST GAME - CHASE GAME

Dags att skapa vårt första spel. Extremt enkelt och det kan knappt räknas som spel då det inte går att vinna eller förlora men man kommer kunna få poäng och det kommer finnas ett utmaningsvärde i programmet.

Game Outline:

Vad vill vi att spelet skall göra?

- Styra med musen.
- En bild skall slumpmässigt spawna på skärmen.
- Klickar man på bilden med musen skall man få poäng.

Pseudokod:

- Kod för att styra med musen
- Ladda bild
- Random funktion för att kunna slumpmässigt visa bilden på skärmen.
- Visa bilden på skärmen.
- Ta bort gamla bilden så det bara finns en bild åt gången.
- Timer för hur länge bilden skall synas på skärmen innan den spawnar om.
- Hitdetection för om musen är över bilden och spelaren klickar på vänster musknapp samtidigt.
- Poäng skall sparas i en variabel om spelaren lyckas klicka på bilden.
- Poängen skall visas på skärmen.

1. Börja med att skapa en liten bild i Photoshop. Den skall bara vara 25 x 25 pixlar stor.
2. Starta ett nytt project i Visual Studio och döp det till "Chase Game" eller liknande.
3. Dags för lite kodning. Öppna Game1.cs det är här vi skall skriva in all vår kod för det här spelet.

Precis under

```
GraphicsDeviceManager graphics;  
SpriteBatch spriteBatch;
```

Skriv in följande kod:

```
Random rand = new Random();  
Texture2D squareTexture;  
Rectangle currentSquare;  
int playerScore = 0;  
float timeRemaining = 0.0f;  
const float timePerSquare = 0.75f;  
Color[] colors = new Color[3]  
{  
    Color.Red,  
    Color.Green,  
    Color.Blue  
};
```

4. Scrolla ner till Initialize delen och precis FÖRE
`base.Initialize();`

Skriv in följande:

```
this.IsMouseVisible = true;
```

Detta gör givetvis så att vi kan använda oss av musen.

5. I Solutions Explorern till höger leta reda på den delen som heter
`ChaseGameContent(Content)` eller liknande lite beroende på vad ni döpte ert spel till.
Höger-klicka och välj `Add – New Folder` och döp den till `Textures` eller `Images`.
Höger-klicka sedan igen och välj `Add – Existing Item` och leta reda på er bild ni
gjorde tidigare. Om ni sedan markerar er bild i listan under er `Textures` folder så får ni
upp en hög med `Properties` för bilden (dyker det inte upp Höger-Klicka och välj
`Properties`). På den rad som heter `Asset Name` döper ni er fil till `SQUARE`.

Detta gör att vi kan länka till just den fil vi precis infogade genom `Asset Namnet`.
I delen som heter `LoadContent()` precis efter `spriteBatch` lägg till följande

```
squareTexture = Content.Load<Texture2D>(@"Textures/SQUARE");
```

6. Nu skall vi lägga till kod så att bilden spawnar om och det gör vi i `Update()` delen så
leta reda på den i er `Game1.cs`.

Precis efter

```
base.Update(gameTime)
```

Lägg till:

```
if (timeRemaining == 0.0f)
{
    currentSquare = new Rectangle(
        rand.Next(0, this.Window.ClientBounds.Width - 25),
        rand.Next(0, this.Window.ClientBounds.Height - 25),
        25, 25);
    timeRemaining = timePerSquare;
}

MouseState mouse = Mouse.GetState();

if ((mouse.LeftButton == ButtonState.Pressed) &&
    (currentSquare.Contains(mouse.X, mouse.Y)))
{
    playerScore++;
    timeRemaining = 0.0f;
}
timeRemaining = MathHelper.Max(0, timeRemaining -
    (float)gameTime.ElapsedGameTime.TotalSeconds);
this.Window.Title = "Score : " + playerScore.ToString();
```

Detta gör även så vi kan få poäng och visar det uppe i vänstra delen av fönstret. Inte så snyggt men vi skall fixa en bättre score i senare spel.

7. Den blå bakgrunden är ganska ful så ändra den till grå. Vi har redan gått igenom detta tidigare så det fixar ni själva.
8. Strax klart för testning och labbning vi måste bara lägga till kod som rensar displayen så det inte finns massor med bilder samtidigt. Så precis efter där ni ändrade bakgrundsfärgen på fönstret lägger ni till följande kod.

```
spriteBatch.Begin();
spriteBatch.Draw(
    squareTexture,
    currentSquare,
    colors[playerScore % 3]);
spriteBatch.End();
```

Yay! Vårt mini-spel är klart. Inte illa för vara ert första lilla spel.