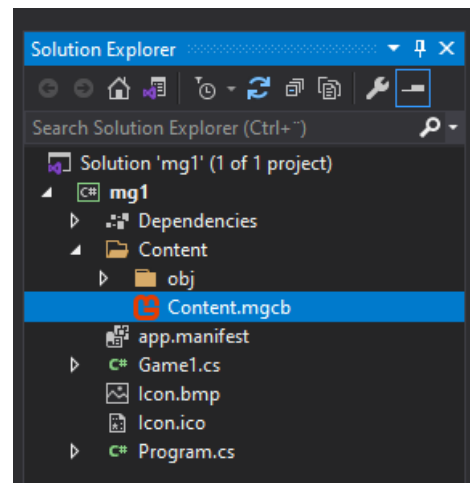


MonoGame - Lägga in Bild

Ladda in bilden i projektet via content editorn, ett program som konverterar filer för att kunna användas i MonoGame-projektet.

1. Öppna projektet där du vill lägga in bilden.
2. Högerklicka på **Content.mgcb** i **Solution Explorer** och välj sedan **Open With**.



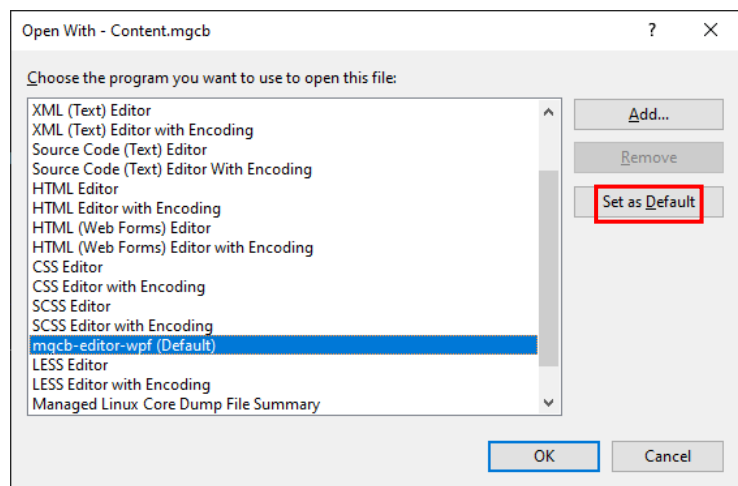
Figur 1

3. Hitta **mgcb-editor-wpf** i listan.

Markera den och välj **Set as Default**.

*Nästa gång ska det räcka att dubbelklicka på **Content.mgcb**, se figur 1.*

Klicka på **Ok** för att öppna editorn.

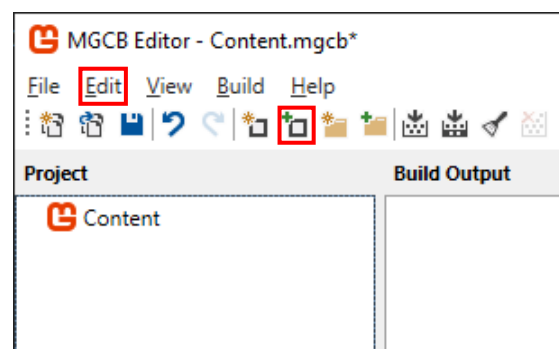


Figur 2

4. För att välja bilden i editorn finns olika alternativ, exempelvis:

Knappen **Add Existing Item** i verktygsfältet,

eller menyalternativet **Edit > Add > Existing Item...**

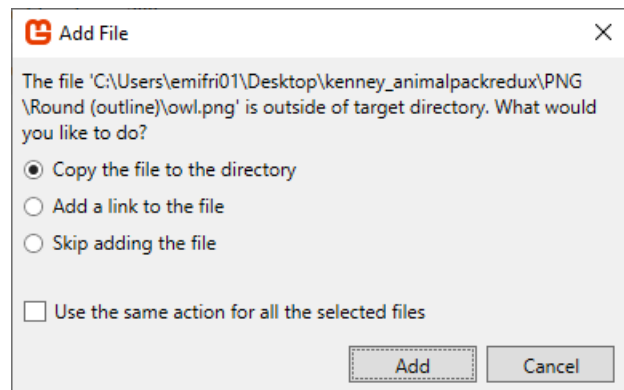


Figur 3

5. Navigera till din bild på datorn och välj att **Öppna** den via dialogrutan i Windows.

När du valt öppna kommer en till dialogruta upp, figur 4, klicka på **Add**.

Det förvalda alternativet gör en kopia av bilden inne i projektmappen, de andra alternativen fungerar också.

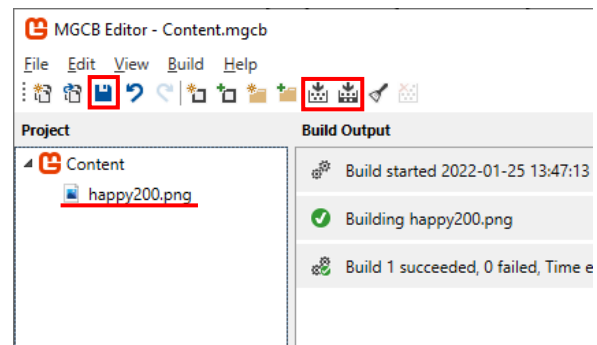


Figur 4

6. Du bör nu se din bild inne i editorn.

Sista steget för att kunna använda bilden är att trycka på en av följande:

Save, Build eller **Rebuild**.



Figur 5

Tillbaka in i Visual Studio. I filken Game1.cs.

7. Deklarera variabler för bildens textur och bildens position.

Jag namnger mina till **bild** och **position**.

Detta görs inne i klassen Game1, men utanför andra metoder, förslagsvis uppe vid de redan fördeklarerade variablerna.



Figur 6

8. Tilldela värden till variablerna. Detta sker i metoden **LoadContent()**.

Ladda in bilden i variabeln, namnet ska stämma med **namnet på filen** i Solution Explorer, men **utan filändelse**.

```
0 references
protected override void LoadContent()
{
    _spriteBatch = new SpriteBatch(GraphicsDevice);

    bild = Content.Load<Texture2D>("happy200");
    position = new Vector2(100, 100);
}
```

Figur 7

Bestäm positionen via en vektor, **enligt (x, y)**. Notera att koordinatsystemet i MonoGame har origo i övre vänstra hörnet av programfönstret och dess y-axel ökar nedåt.

9. Instruera programmet att rita ut bilden, detta sker i metoden **Draw()**.

Lägg till metoderna **Begin()**, **Draw()** och **End()**.

Lägg in variablerna för **bilden** och dess **position** som argument, samt **färgen** som ska användas.

```
0 references
protected override void Draw(GameTime gameTime)
{
    GraphicsDevice.Clear(Color.CornflowerBlue);

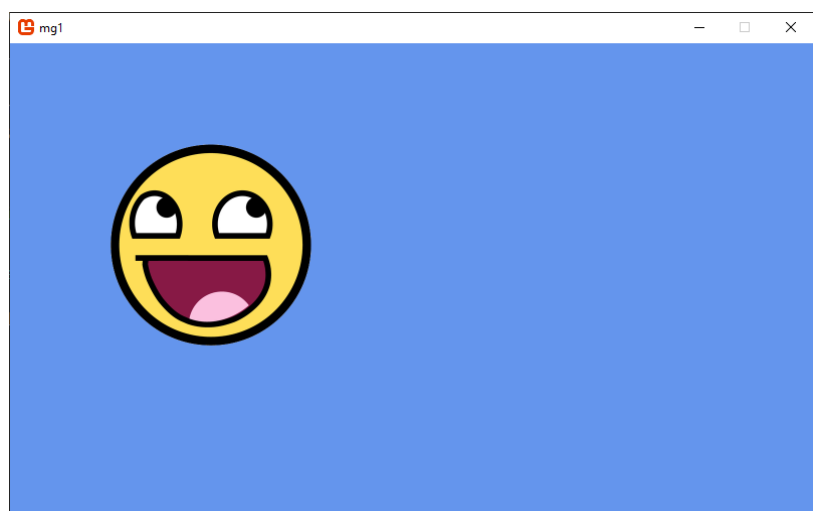
    _spriteBatch.Begin();
    _spriteBatch.Draw(bild, position, Color.White);
    _spriteBatch.End();

    base.Draw(gameTime);
}
```

Figur 8

10. Bilden bör nu dyka upp i programfönstret när man startar programmet.

Tips: Bakgrundsfärgen kallas CornflowerBlue, detta kan ändras, se figur 8.



Figur 9