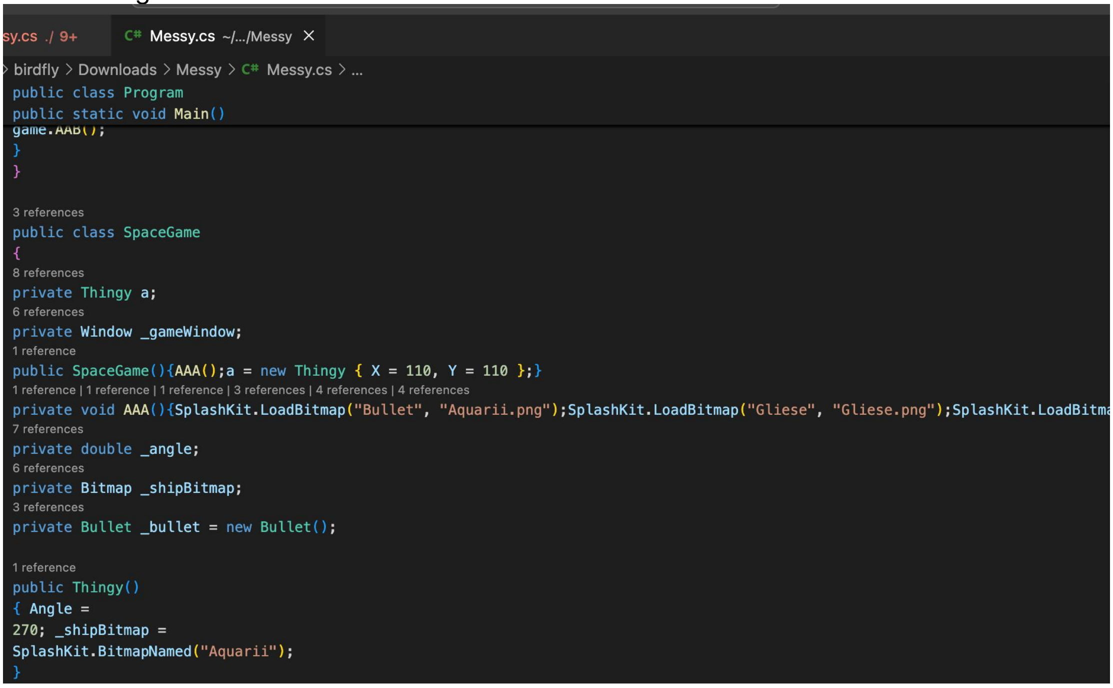
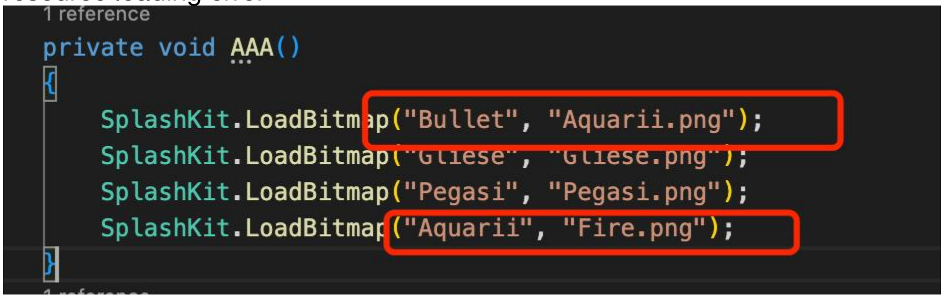
no indenting: code is hard to read



sy.cs ./ 9+
C# Messy.cs ~/.../Messy ×
birdfly > Downloads > Messy > C# Messy.cs > ...

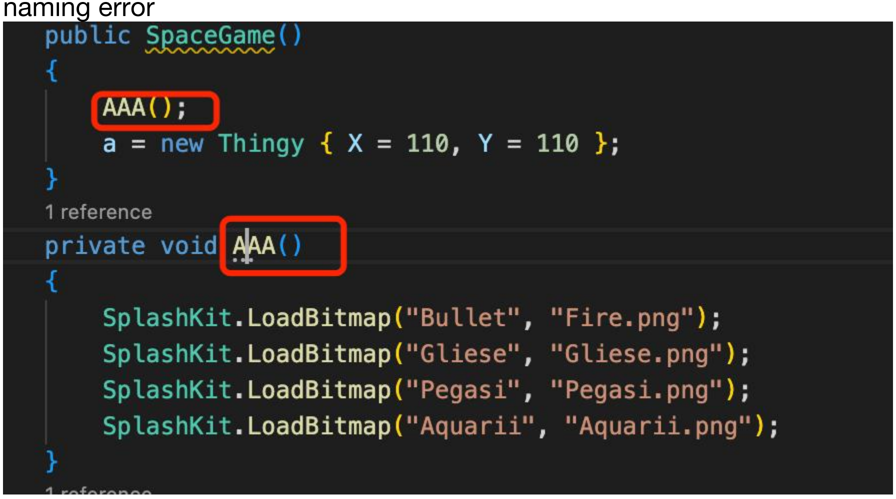
```
public class Program
public static void Main()
game.AAB();
}
}

3 references
public class SpaceGame
{
8 references
private Thingy a;
6 references
private Window _gameWindow;
1 reference
public SpaceGame(){AAA();a = new Thingy { X = 110, Y = 110 };}
1 reference | 1 reference | 1 reference | 3 references | 4 references | 4 references
private void AAA(){SplashKit.LoadBitmap("Bullet", "Aquarii.png");SplashKit.LoadBitmap("Gliese", "Gliese.png");SplashKit.LoadBitma
7 references
private double _angle;
6 references
private Bitmap _shipBitmap;
3 references
private Bullet _bullet = new Bullet();

1 reference
public Thingy()
{ Angle =
270; _shipBitmap =
SplashKit.BitmapNamed("Aquarii");
}
```

resource loading error



```
1 reference
private void AAA()
{
    SplashKit.LoadBitmap("Bullet", "Aquarii.png");
    SplashKit.LoadBitmap("Gliese", "Gliese.png");
    SplashKit.LoadBitmap("Pegasi", "Pegasi.png");
    SplashKit.LoadBitmap("Aquarii", "Fire.png");
}
```

naming error



```
public SpaceGame()
{
    AAA();
    a = new Thingy { X = 110, Y = 110 };
}
1 reference
private void AAA()
{
    SplashKit.LoadBitmap("Bullet", "Fire.png");
    SplashKit.LoadBitmap("Gliese", "Gliese.png");
    SplashKit.LoadBitmap("Pegasi", "Pegasi.png");
    SplashKit.LoadBitmap("Aquarii", "Aquarii.png");
}
```

Should the ultimate goal of a software project be code that works, or is there more to it than that?

A good software project not only requires its code to run but also to be easy to maintain and extend. This means the code needs to be written in a way that is easy for humans to understand, and its design should follow a series of principles such as OCP, LSP, and others.

What made the code hard to read?

1. No formatting, no indenting;
2. Confusing naming convention;
3. Code redundancy;
4. Logic confusion;

If this were a team project, how important to do think coding standards would be?

It is very important to do think coding standards before start the project, Only when all members reach a consensus can the project proceed. In this way, there will not be multiple coding styles in the project, and members can reach a consensus on boundaries.

What does this mean for software development in general?

This means that projects will be built on the same development style, reducing maintenance and extension work

Do a quick search on the idea of code refactoring, how does this relate to what you have been doing?

When developers use different styles to build a project, the project will be difficult for newcomers to understand and adapt to. When the project needs to be refactored, it will be a difficult task and require more refactoring work.