Exercise – Class Diagrams

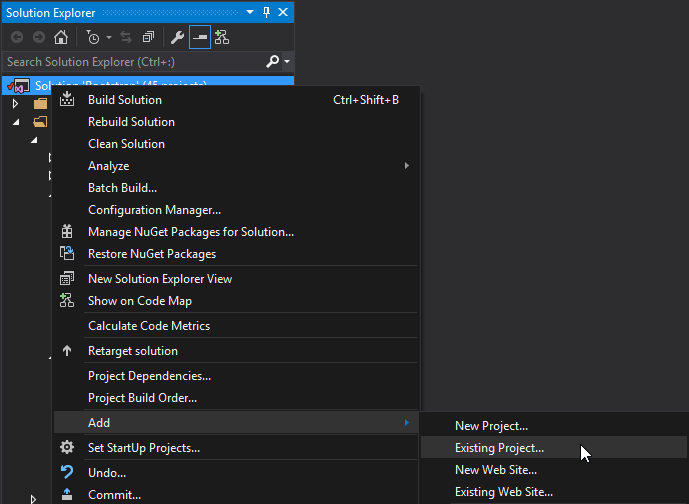
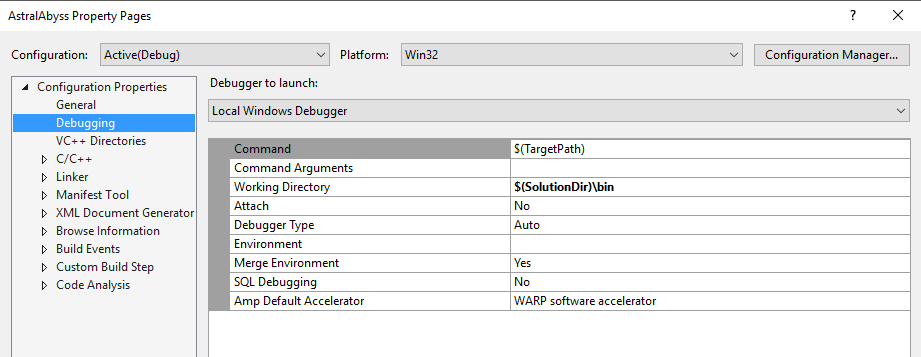
Introduction:

In this exercise (and in the exercises for the next several sessions) we are going to analyse a simple game so that we can document its design using UML diagrams. We’ll begin in this exercise by creating a Class Diagram for the game Astral Abyss.

Set Up:

The Astral Abyss project is available on the Resources page for this subject.

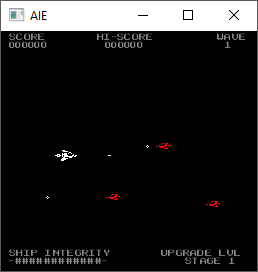
The game is provided as a project that will link into the *aieBootstrap* solution. If you do not yet have a copy of *aieBootstrap*, you will need to download that from this github repository: <https://github.com/AcademyOfInteractiveEntertainment/aieBootstrap>

1. Download *AstralAbyss.zip* from the *Resources* page for this subject
2. Extract the zip file to your computer. (A good place to extract it would be to the bootstrap solution folder)
3. Open the *aieBootstrap* solution
4. In Visual Studio, add the *AstralAbyss* project to the solution.  
   In the *Solution Explorer*, right-click on the solution and select *Add -> Existing Project*   
   
5. Open the properties for the *AstralAbyss* project and ensure the debug *Working Directory* is set to **$(SolutionDir)bin\**  
   
6. Lastly, we need to copy the images and fonts this project uses to the solution’s *bin* folder.

In the AstralAbyss project folder you will find a *bin* folder containing images and fonts. Move these into the **$(SolutionDir)\bin** folder.   
  
If your game launches and you cannot see anything drawn, you have likely copied the resources to the wrong folder.

Once you have set up the project, set it as the active *Start Up Project*, compile and then run the project.

You should be able to launch and play the *Astral Abyss* game without errors.



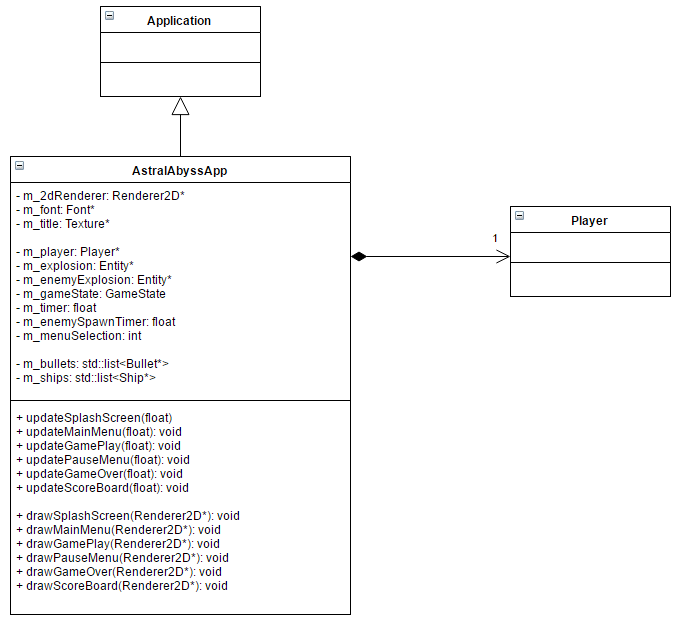
Exercise:

Either individually or in groups, create a class diagram that accurately describes the architecture of this game.

Ensure you list all classes, and the important member variables and functions of each class.

You will probably find it much quicker to draw your design on paper. Once you have finalised your design you may then wish to transfer it to a drawing program like <http://draw.io>.

Below are a few classes to get you started with your class diagram:



As you draw your diagram, try not to focus too much on listing all of the variables and functions of each class.

Rather, focus on defining the relationships between classes, like inheritance, composition, and aggregation.

Fill in specific details like whether or not a relationship is uni- or bi-directional, and add the multiplicity values to your relationships.