

## Chapter 11: Separate Compilation

Generally, classes are split into two files:

1. Header file: contains member variables and function declarations (no implementation). Also called the **interface** for a class. Header files end with **.h**. Functions are fully documented (using comments) to indicate how the function works.
2. Implementation file: contains the definitions/implementation of the functions declared in the header file. Our implementation files will end with **.cpp**.

Header file:

- Any program that uses a header file must contain an **include** directive. For example, a User-defined "Rat.h" file would be included in the implementation file "Rat.cpp" as following:  
`#include "Rat.h"`
- If a header file is a built-in, predefined file, the header file name is written with angular brackets.  
`#include <iostream>`
- Header files do not need to be compiled

Implementation file:

- Must include the header file (see above)
- Must be compiled, even after a change is made into an "object" file

Driver / Application file:

- Contains the main function to test/use the class
- Needs to include only the header file
- Must be compiled, even after a change is made

Connecting the implementation file and driver file together is called "linking".