Homework 3

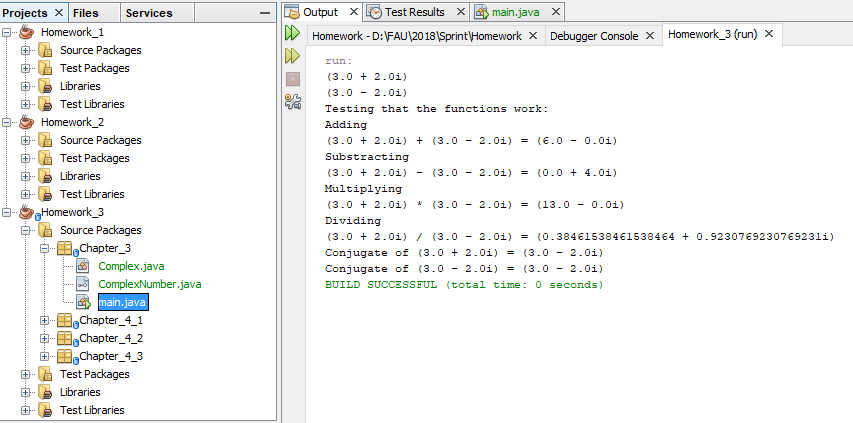
3.1

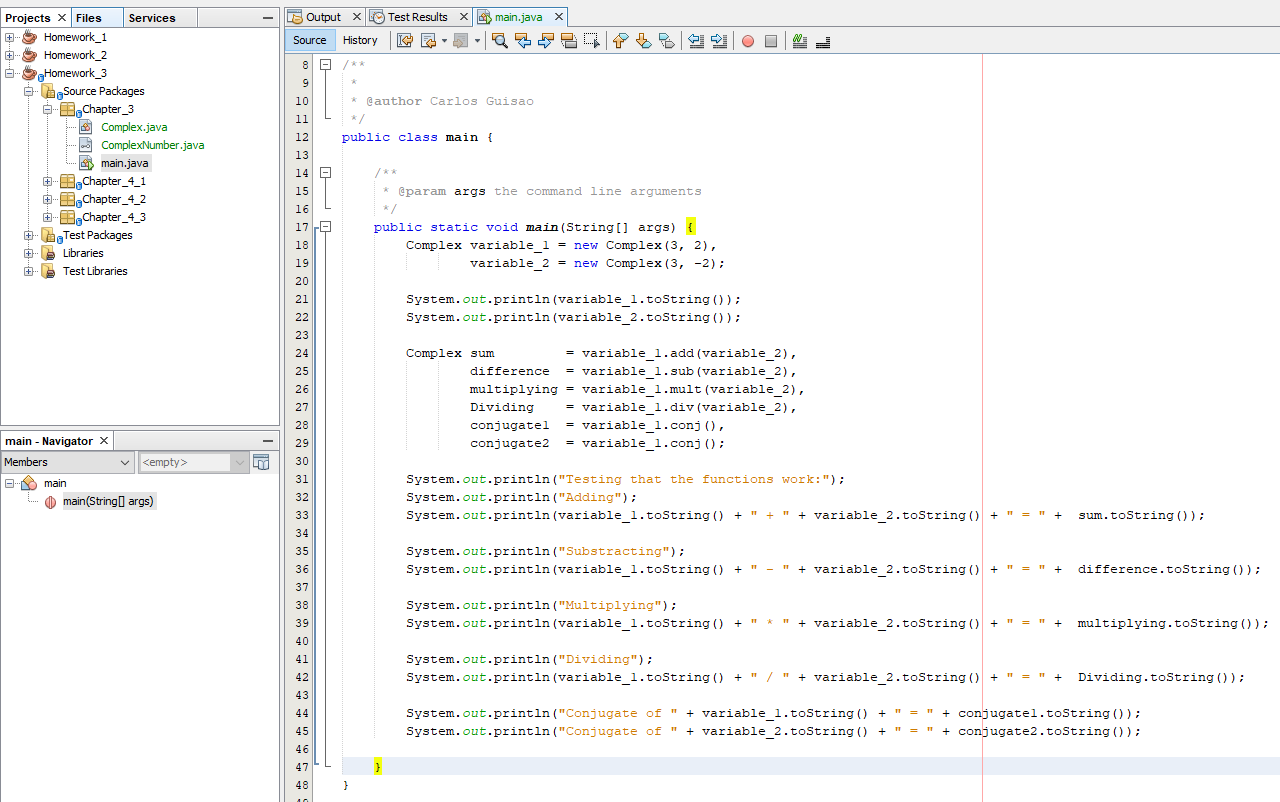
A. Encapsulation is an important principle of OODP because it hides the data implementation by restricting access to accessors and mutators. This allows for code that is reusable and maintainable for both people using the code and designers.

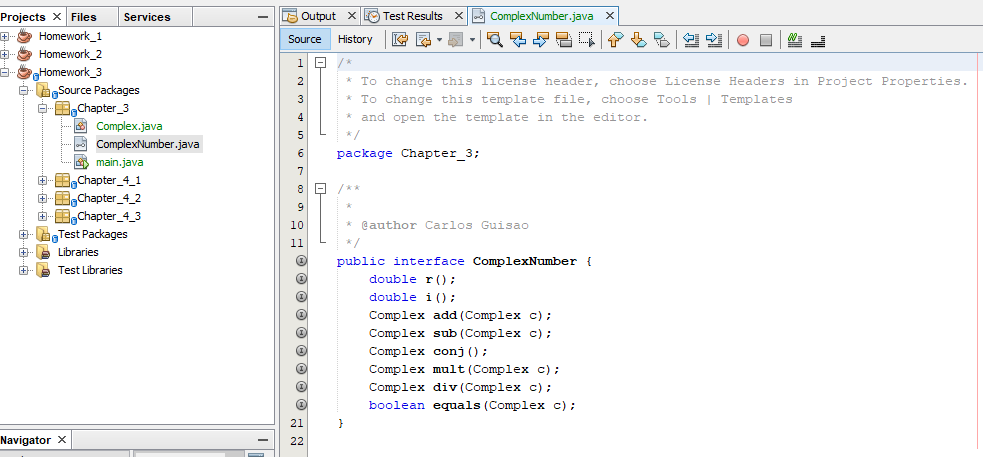
B. It is okay to throw exceptions as part of the contract whenever the precondition fails or is not fulfilled. One example can be removing an elements at the head of a linked list when the list is empty or set to null.

C. Side effects should be avoided because they can cause unnecessary changes to an object and or its state that you may or may not be aware of.

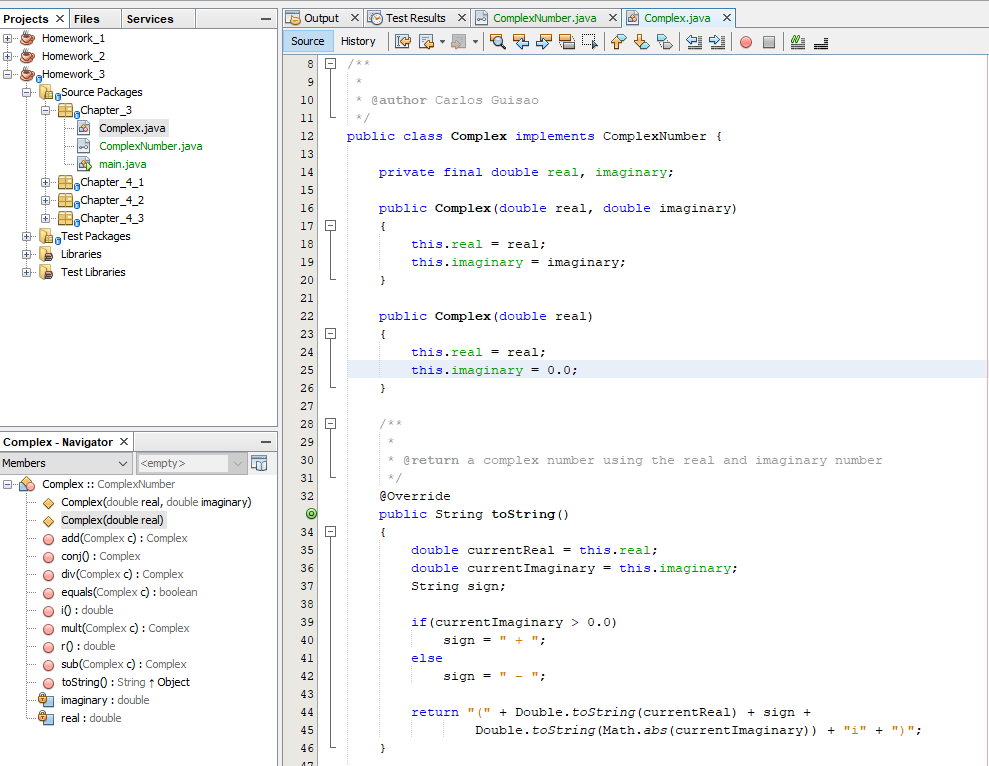
3.2

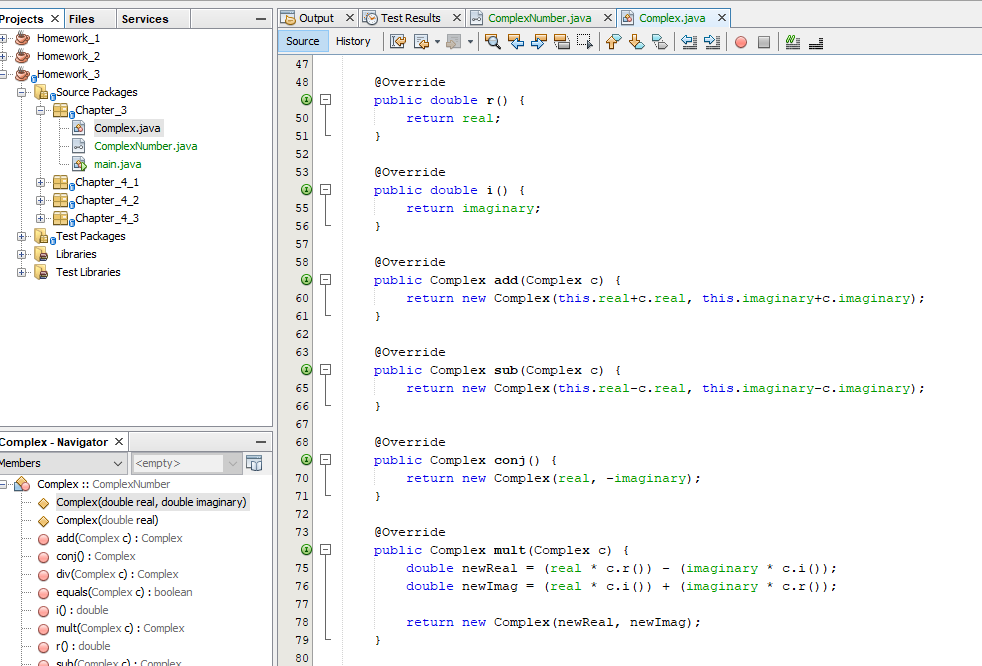


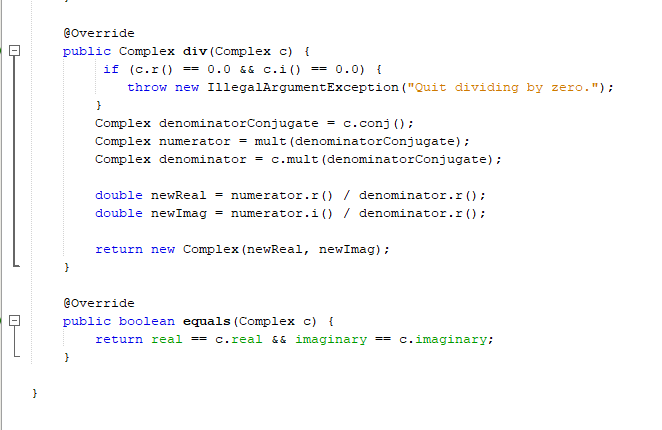




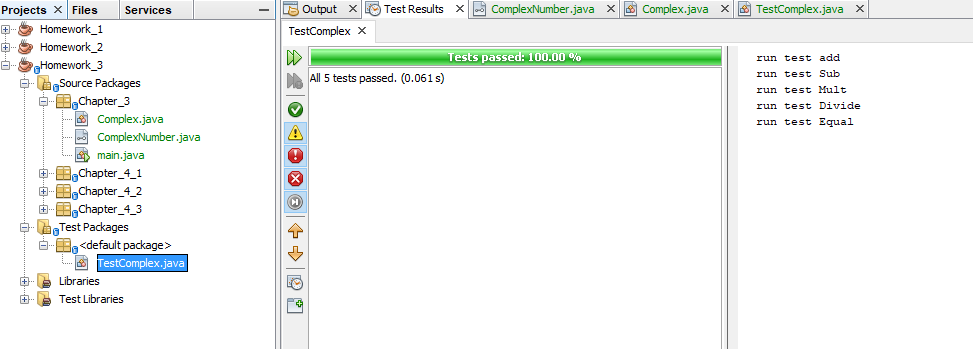
Complex Class

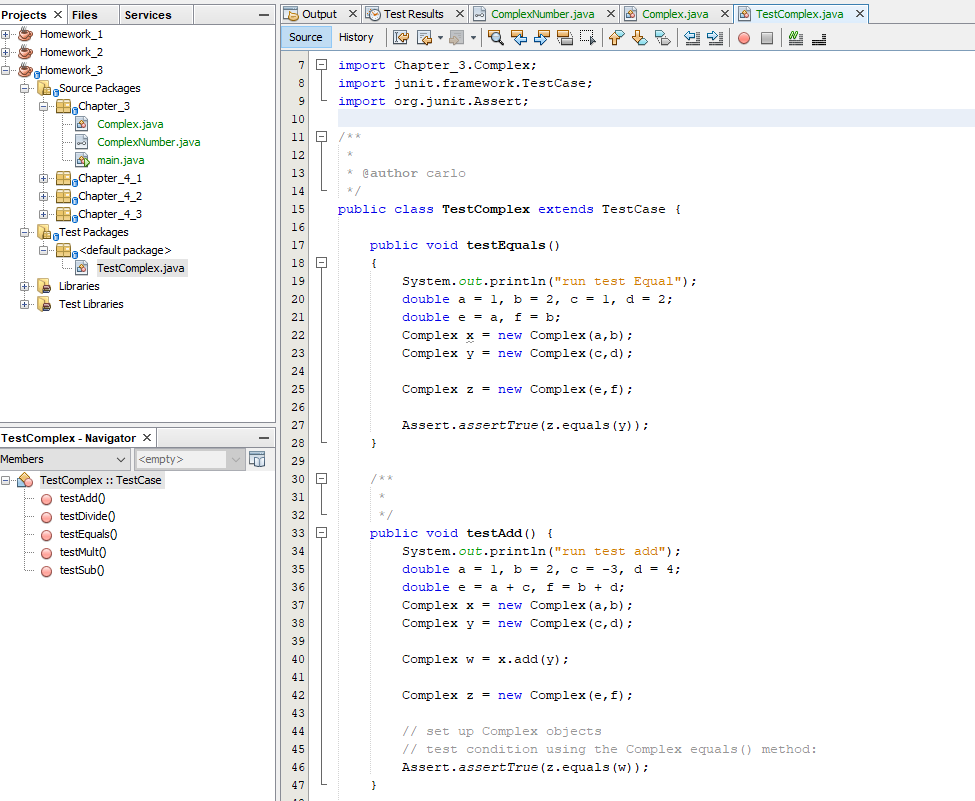


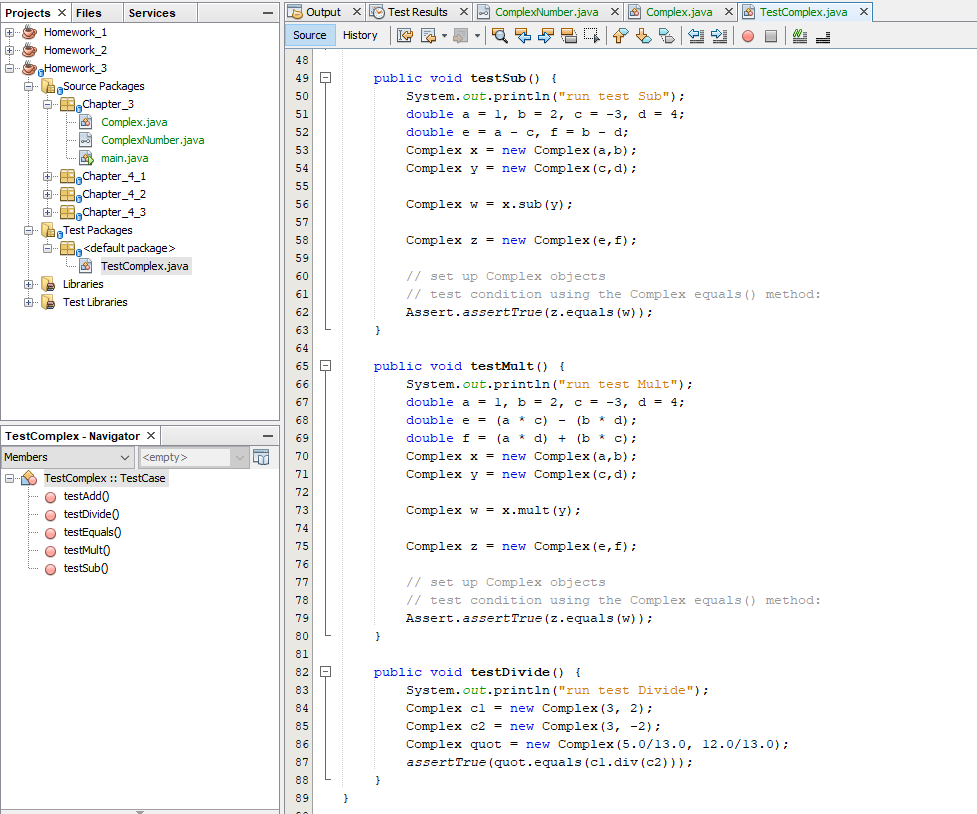




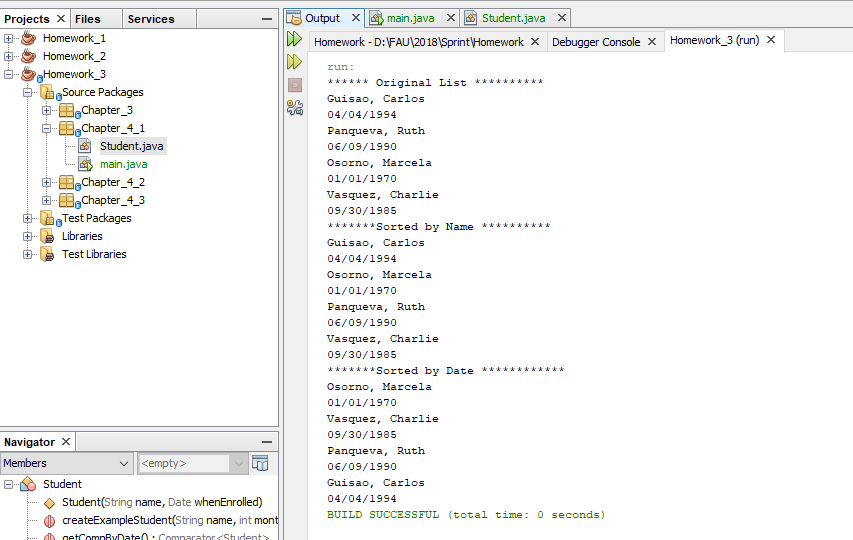
Unite test



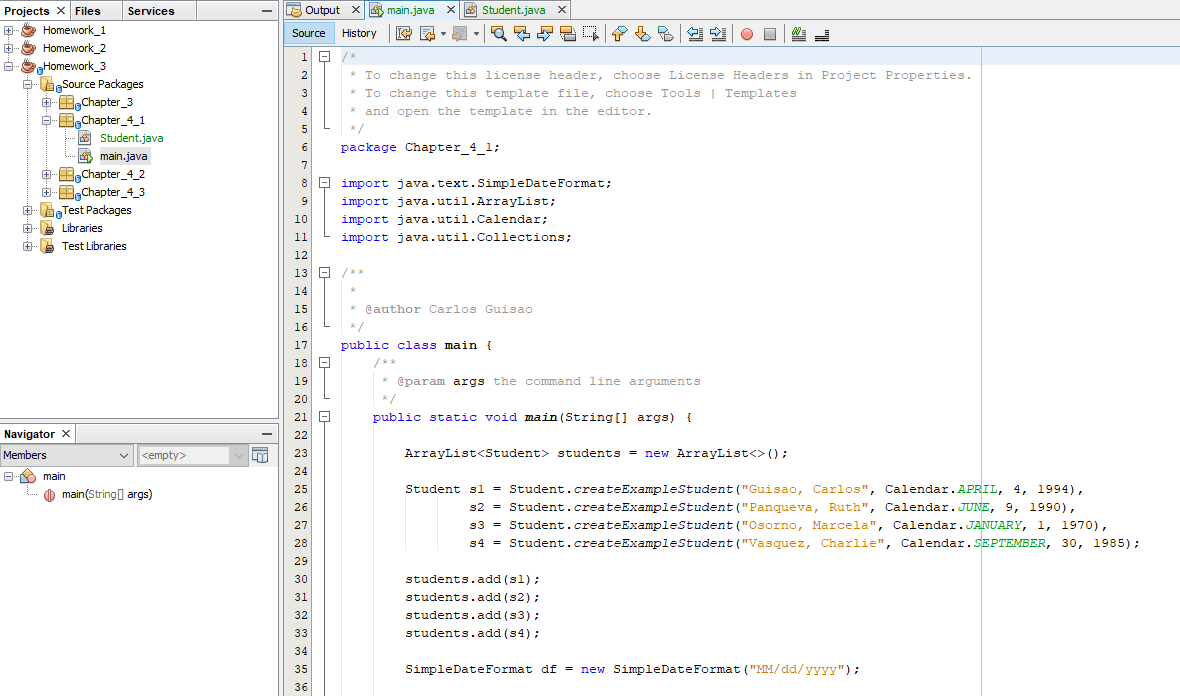




4.1 Student Class

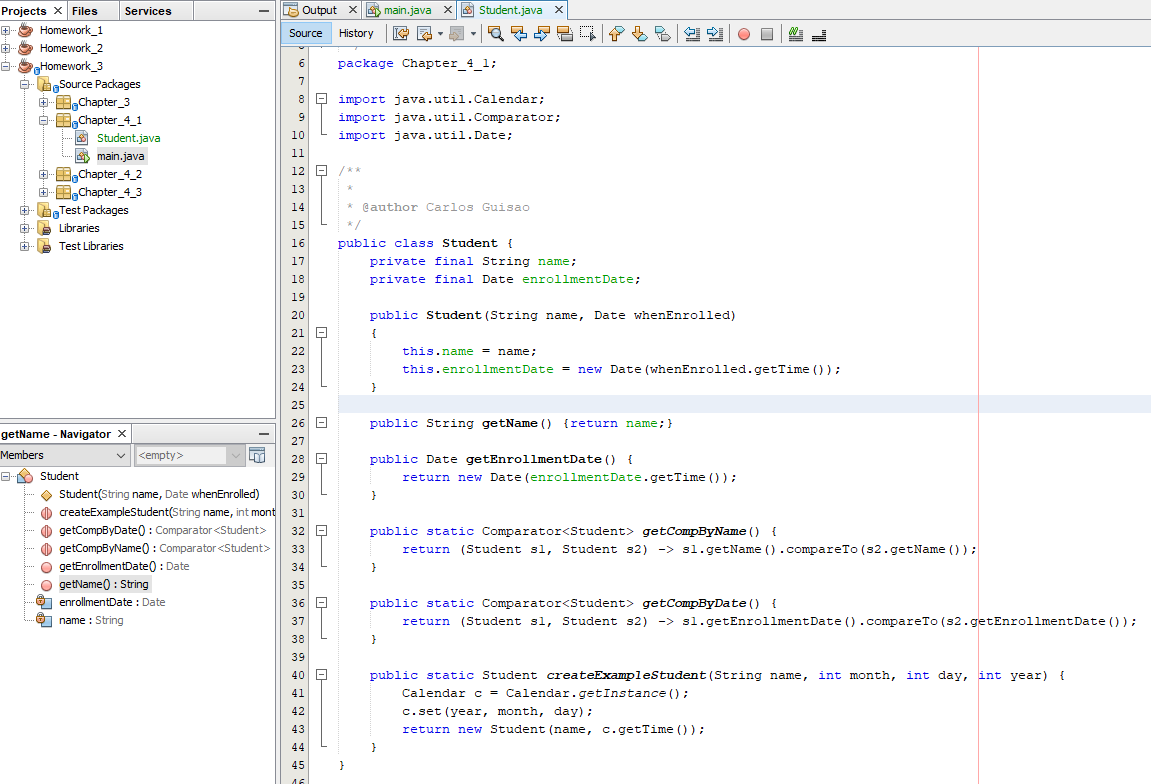


Main Class

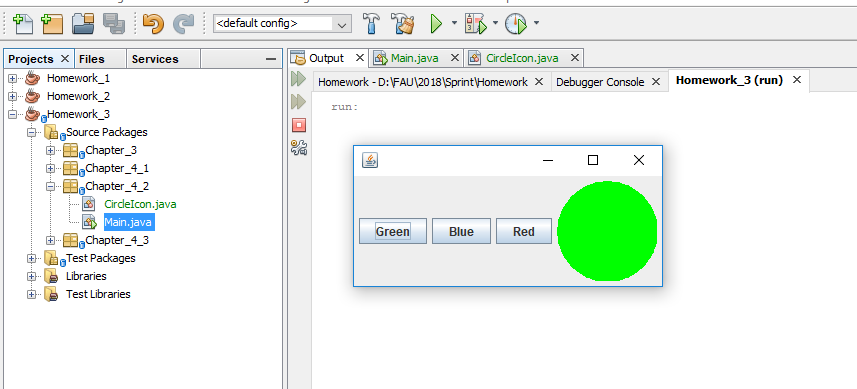


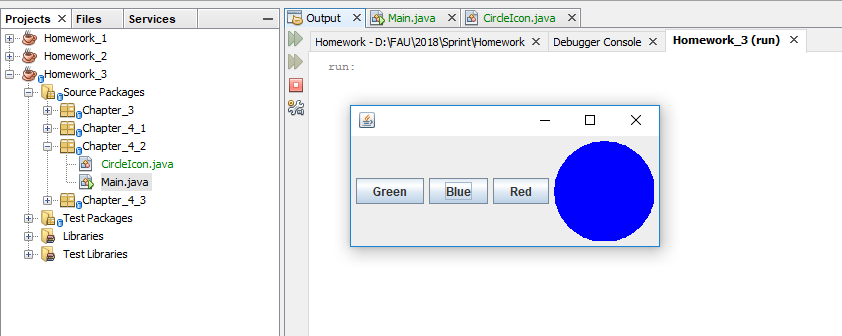


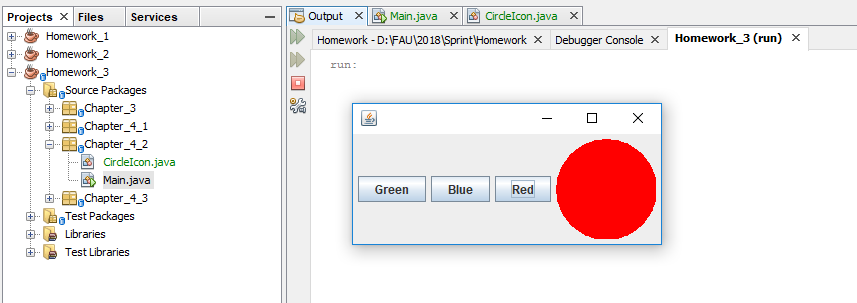
Student Class



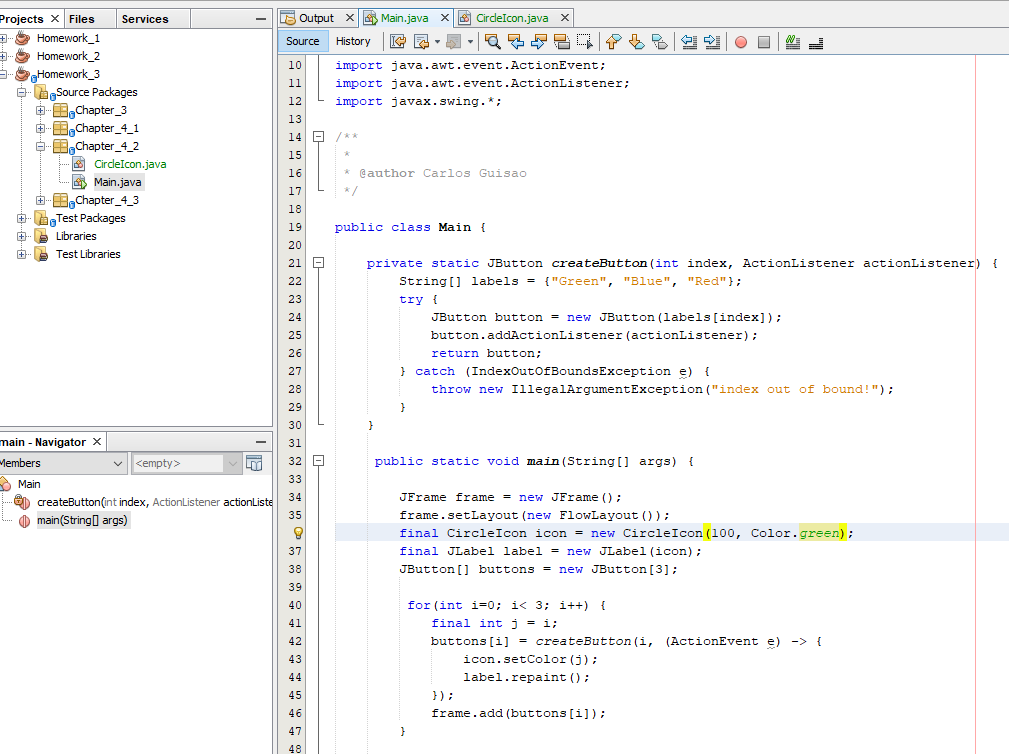
4.2

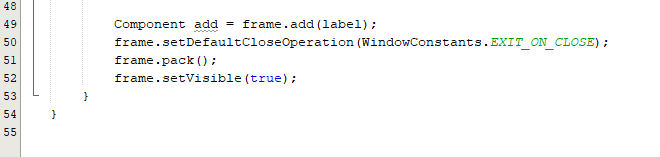




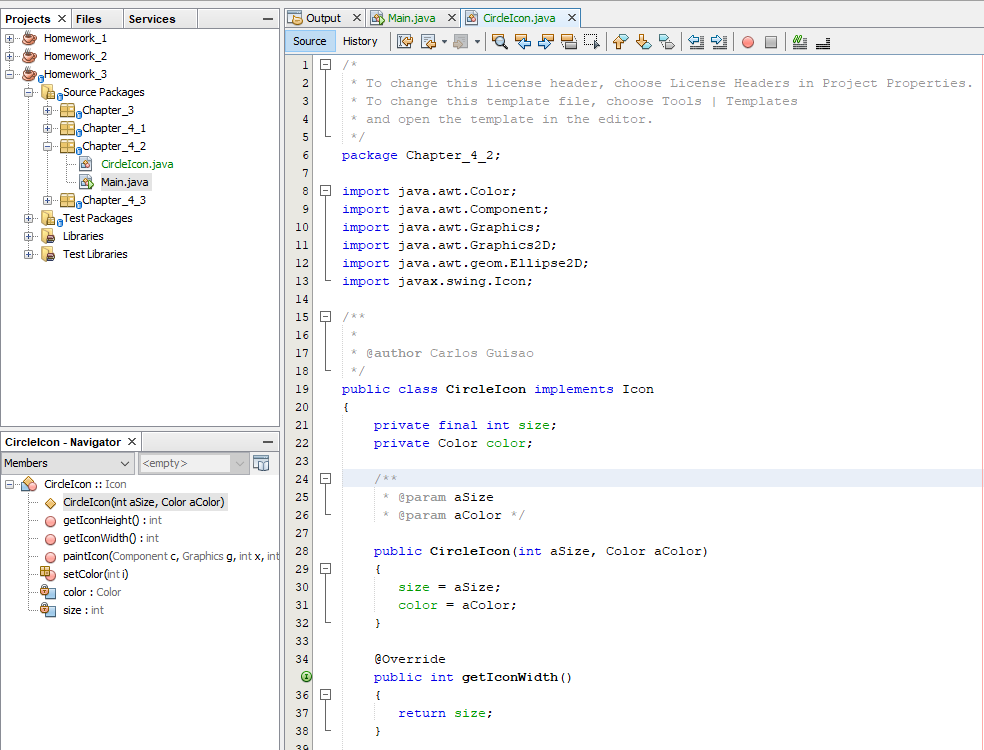


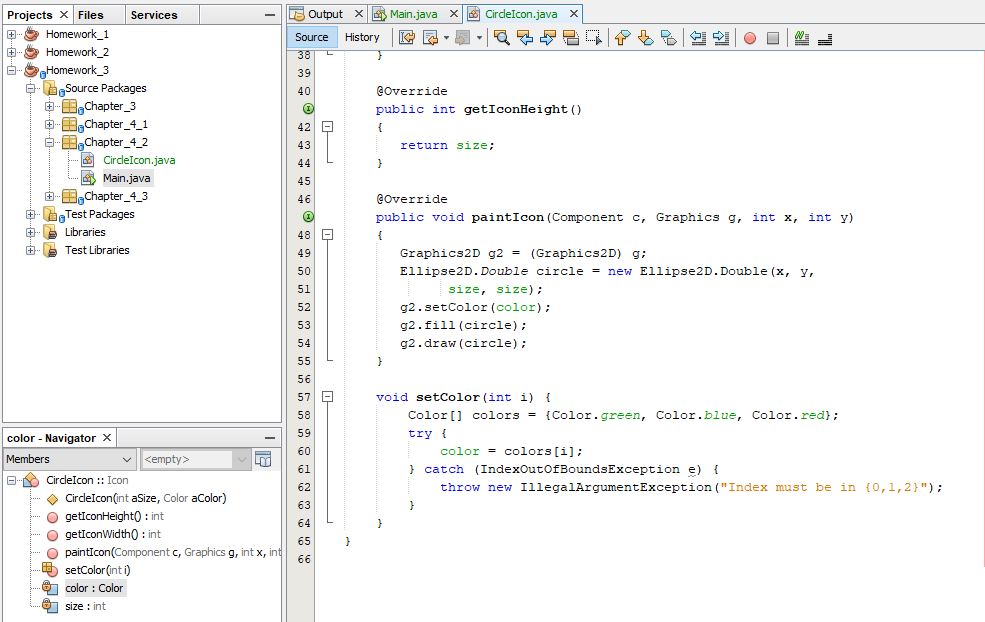
Main



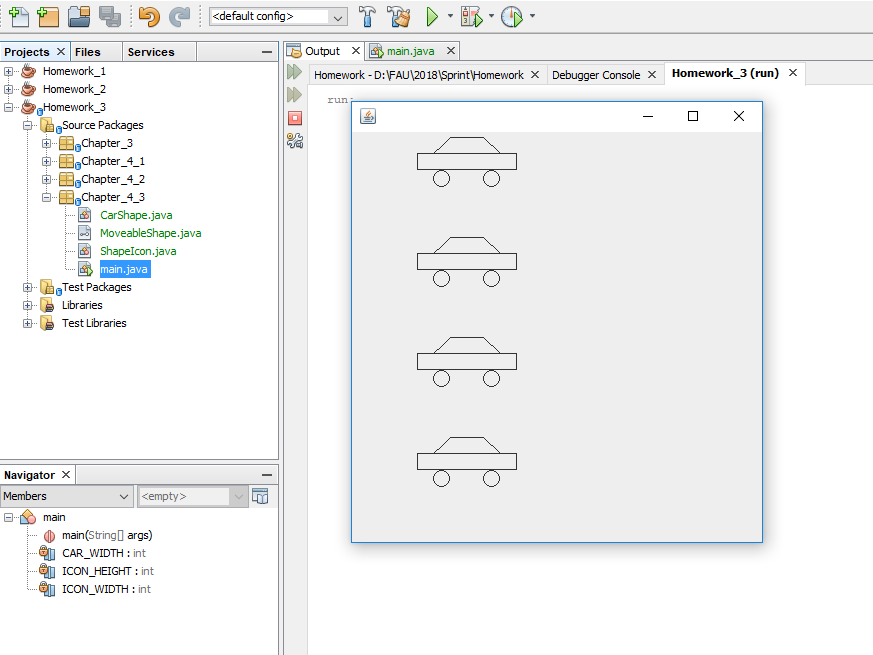


CircleIcon Class

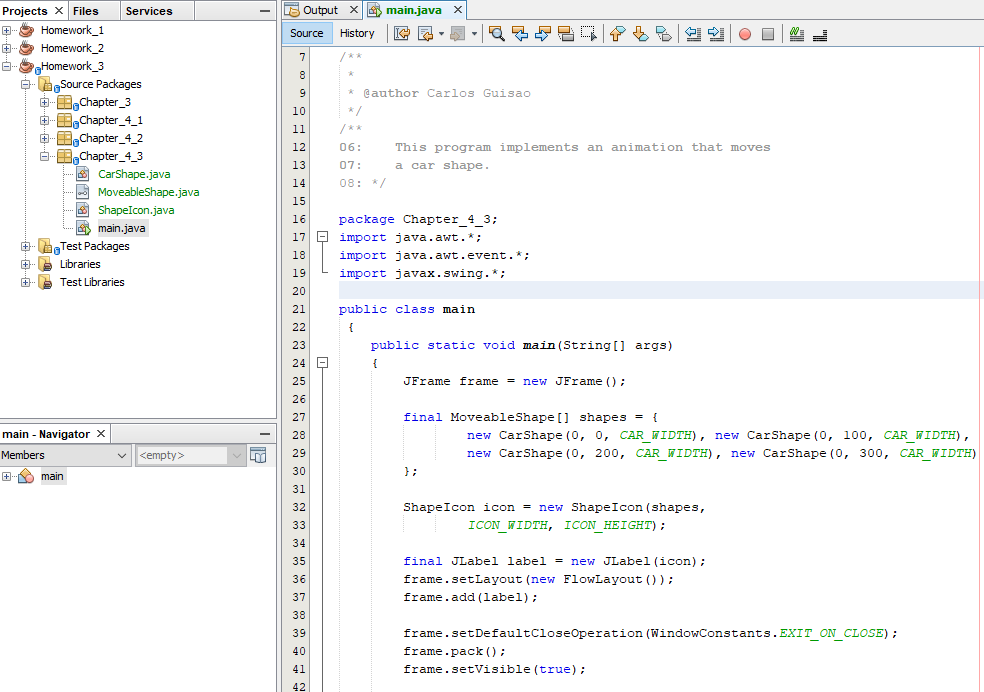


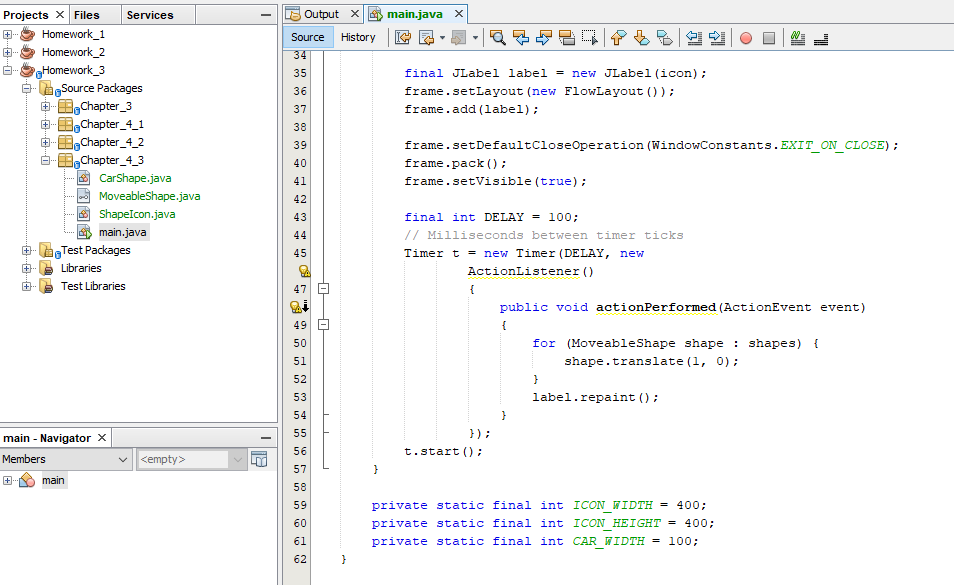


4.3

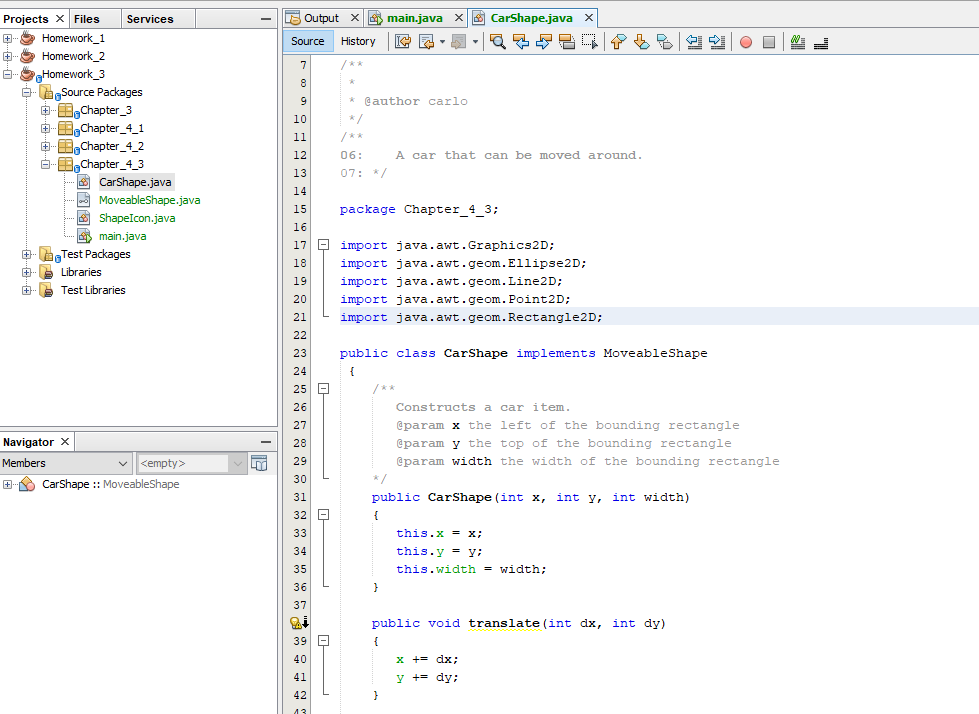


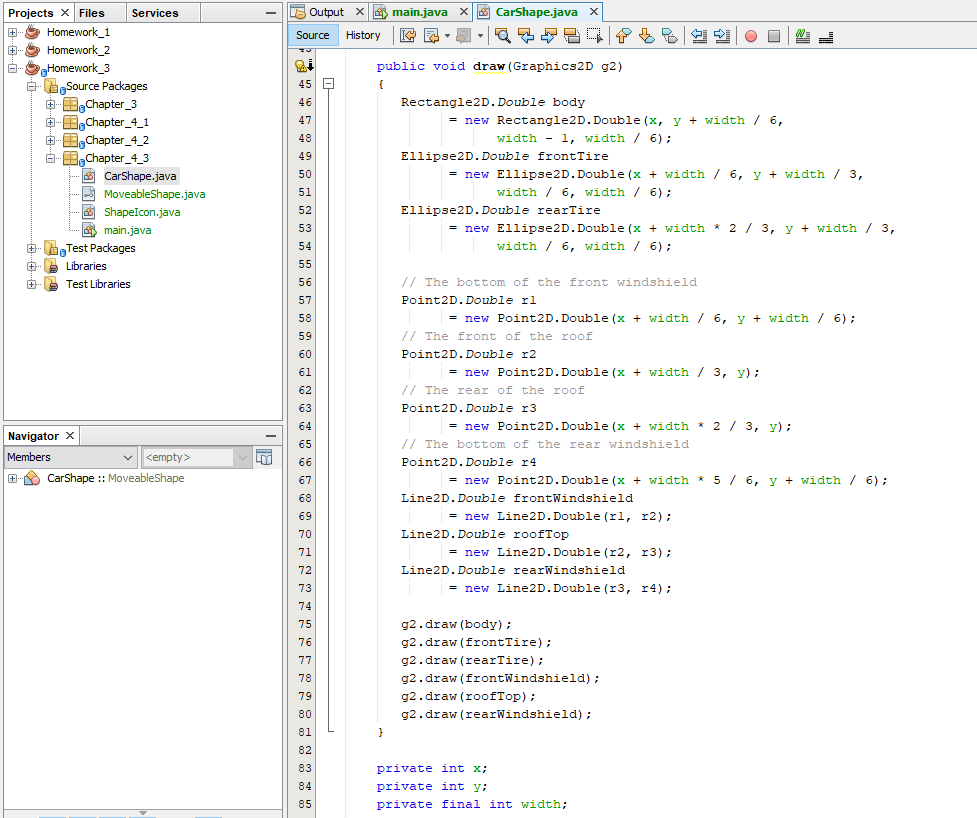
Main



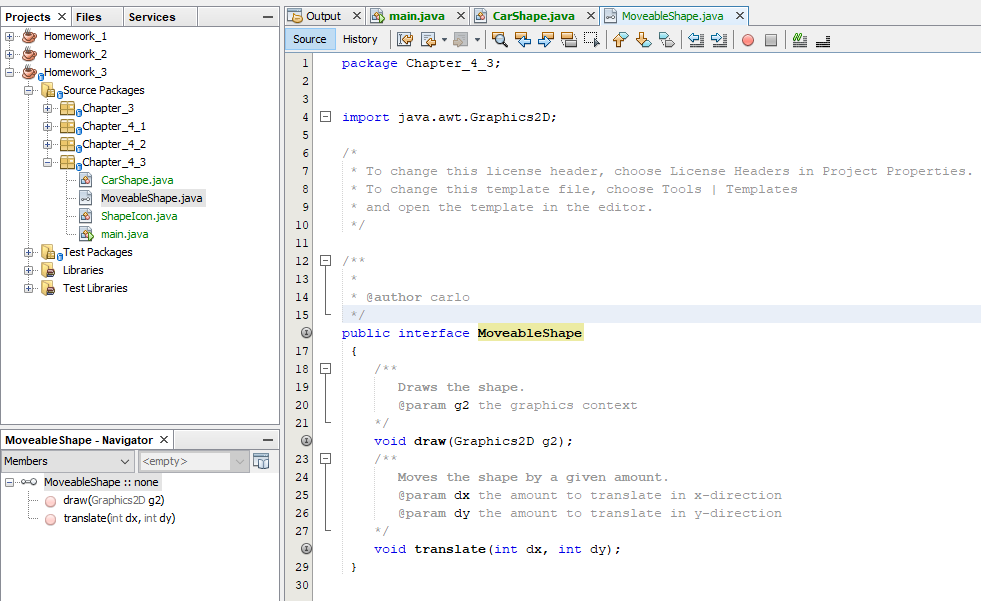


CarShape Class





MoveableShape Class



ShapeIcon Class

