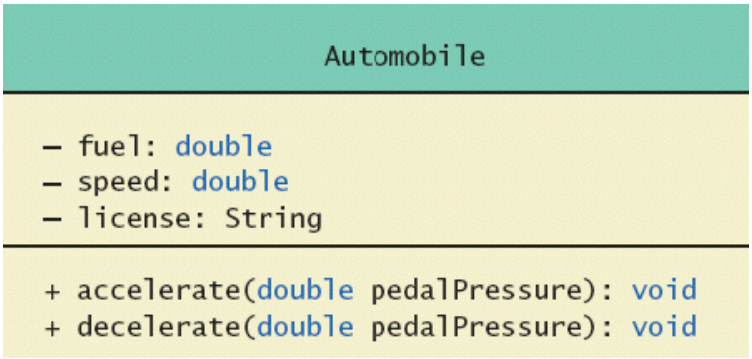
Items to study for COMP 170 Exam 1, Thursday, February 28 – in general, Weeks 1-5 of the course, Savitch Chapters 1 through 5, through loops; several of these topics / items will appear on the Exam:

- **Java constants and literal values like 2, 3.14159, and "abc"; also Java variables of various types**
- **Java numeric assignment statements and expressions using operators like +, -, *, /, and %**
  - **Be able to distinguish between integer division and decimal division**
  - **Remember that a%b means the modulus or remainder after dividing a by b**
  - **Remember that *, /, and % are done before + and - unless there are parentheses**
  - **Remember about automatic conversions, for example from char to int, int to double, …**
- **The difference between System.out.print and System.out.println, and how to use each of them**
  - **How to print a blank line using System.out.println: System.out.println();**
  - **What is printed for each data type; in particular, chars print their character value if you print them with System.out.print/println, just as though they were string values**
- **Strings, string indexing, string methods like length, substring, indexOf, …**
  - **String assignments and using + with strings (and "adding" strings to other types)**
  - **What are valid indexes – where do they start and stop? [start at 0, go to length - 1]**
  - **The two different versions of substring and what they mean (1 parameter vs. 2)**
  - **Escape characters used with Strings (and chars): \n, \t, \", \', \\, …**
- **If and else statements, if logical conditions, and which statements are controlled by each**
  - **Using curly braces to delimit the statements under control of an if/else if/else**
  - **Use of *else if* – what it means and how it goes together with other parts of an if**
  - **Nesting of if statements inside each other (and also nesting inside / around loops)**
- **While, for, and do-while loops and the conditional/logical expressions that they depend on**
  - **Understand how many times a while/for/do-while loop executes based on its condition**
  - **Understand what happens inside the "body" of a loop vs. before or after it executes**
  - **Possibly for-each loops – may be useful, but not required**
- **For boolean expressions in general, know how to use relational operators like ==, <, >=, etc, and how to use logical operators like && and ||**
  - **Remember that && and || are done after all other numeric and relational operators unless there are parentheses**
  - **Remember that && and || "short-circuit" and understand what that means**
- **Reading keyboard input using Scanner**
  - **The requirement for import java.util.Scanner; (or java.util.*;) at the top of your program**
  - **The need to create a Scanner object before reading input:**

    Scanner keyboard = new Scanner(System.in);
  - **The various methods you can use with a Scanner object:**
    - **nextInt, nextDouble, nextBoolean, … → read and return values of those types**
    - **next → read a single "word", delimited by whitespace (space, tab, Enter, …), and return it as a String**
    - **nextLine → read the rest of the line up to Enter and return it as a String, but without the ending newline character that Enter sends in**

- **With this UML below (answer in practice exam)**
  - **Please write the code for it**
  - **Makes comments in the code you write:**
    - **What are the methods**
    - **What is the class names**
    - **What are the primitive types**
    - **What are the class types**
    - **Which variables and functions are public or private**



Definitions you should know (all in the book):

• **Compiler** • **Encapsulation** • **Algorithm** • **CPU** • **Memory (auxiliary and secondary)** • **Byte and Bit** • **Assembly Language** • **High-Level Language** • **Class and Primitive Types** • **Assignment** • **Operator** • **Type Cast** • **Concatenation Operator** • **Acessors** • **Mutators**	• **Substring** • **Character Set** • **Delimiter** • **Compound Statements** • **Logical Operator** • **Short Circuit Evaluation** • **Enumeration** • **Class** • **Function/Method** • **Infinite Loops** • **Private Variable** • **Public Variable** • **UML Diagrams** • **Static Function**