ICT167 ANS4

Methods calling method:

- Methods invoked outside the object definition requires an objectID attached to the method UNLESS a static method/class method. A method can invoke other public/private methods without the attaching an objectID (object)
  - EG: answer(), seekAdvice(), Math class methods
- BUT a class program that has a main method should be treated as a client program. IE A MAIN method in a class program needs to attach an objectID when invoking other public/private methods.

- Private methods can only be accessed/invoked inside class by another private method *
  - No objectID attached when private method wants to invoke another private method
- Public methods in a class can be accessed/invoked by another class/client *
  - ObjectID attached when want to invoke in client
  - *Consider the effects of static first

Static variables: (static variables are used if it does not belong to any specific object)

- RULE OF THUMB-
  - Never use it since in both assignments I never had to
- A static variable is a variable that stores the same value for ALL objects created and can be accessed by all objects for that class and can be accessed by all objects for that class
- Think: Global variable for every new object that can be accessed by every object
- Changes to the static variable changes the variable for all object unless there is keyword FINAL

- Declare as private although doesn't matter but for information hiding
- Use ClassName.[Variable] to access from another class
- Syntax
  - Public static final int UPPER_LIMIT = 999;
  - Private static int counter;
- Commonly referred to as class variable

Static methods: (static methods are used if it does not belong to any specific object)**

- RULE OF THUMB-
  - NEVER use it UNLESS you must because of client program
  - "Does it make sense to call this method, even if no object has been instantiated yet?" eg: CMtoMM() ← Useful
    };

- Keyword static results in no need to include objectID attached to method when invoking it
- Use ClassName as ObjectID to access from another class (except for same class access)
- Static method CANNOT INVOKE a non-static method of the same class
  - So main method is a static thus must only invoke static methods

- Commonly referred to as a class method
- Syntax
  - Public static Boolean isPositive() {

    };

Top-Down Design: (approach to writing java programming)

- The methodology we need to use to design our Java programs (for this unit)
- Known as divide and conquer/stepwise refinement which means breaking the problem down into smaller steps
- Come up with a few steps at the highest level on how you can solve a problem. Then we break the large steps into smaller steps such the steps are the lowest indivisible steps
- Breaking the program down to the smallest indivisible steps
- Solutions to sub-tasks might be implemented as a private helper method

Testing a complex program:

- The way we should test is to test all the methods but only test one new method at a time
- Driver program:
  - The main method that is embedded in class so that you can run and test the different method available you're the class. Should only test one new method at a time
- The two approaches to testing are top down testing and bottom up testing

Top down testing:

- Where we test method A first before we test method B through the use of a stub
- A stub is a trivial method that does something simple to let the invoking method know that they can successfully communicate together

Bottom up testing:

- Test method B fully first using a driver program. Check whether the output of method B is correct and desired
- Then we test method A which will invoke method B. If there is an error than we can be assured that it comes from method A as we essentially isolated the method

WRAPPER (class) methods:

- Convert primitive type variable ← → class type variable by wrapping up the primitive data type as an object belonging to a class /unwrapping object back to a primitive type variable
  - Allows us to use wrapper methods
- The class type variables may be → Character, Integer
  - Both types have different methods

MATHS (class) methods:

this.: refers to the current calling object