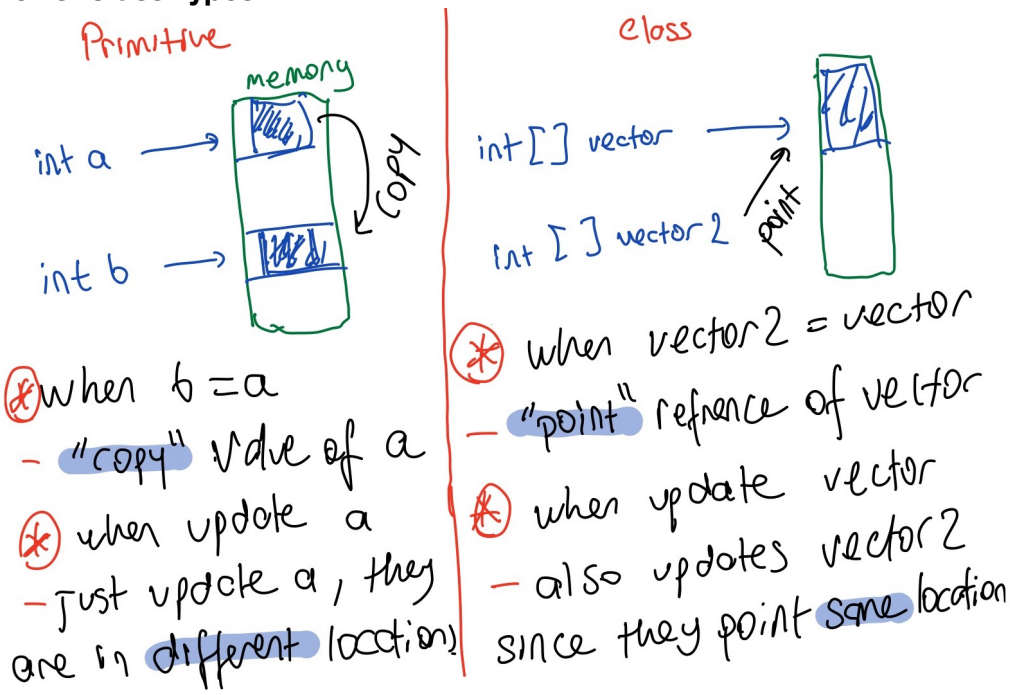
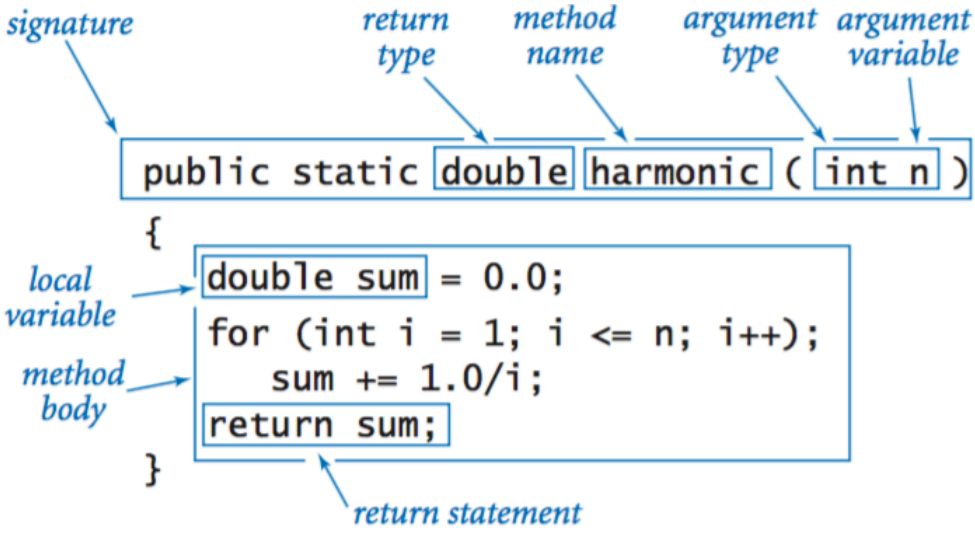
Week5 Notes

Primitive vs. Class Types



Functions

- A function is a block of code that perform certain actions.
- General Syntax:

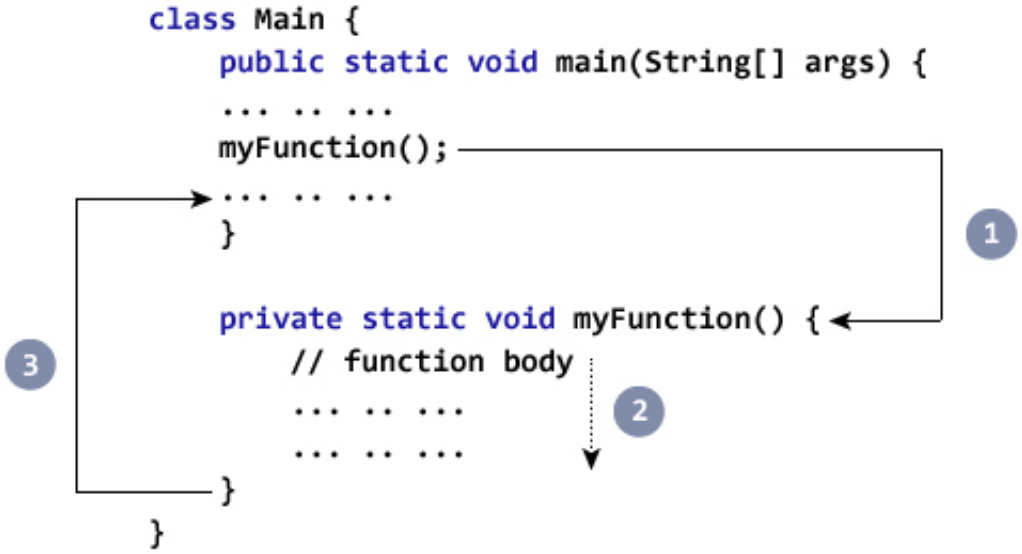


Source: Princeton

- There can be multiple argument variables (i.e., input parameters). You have to define

types of variables. When you call your function, you must give parameters to your function in a same order.

- When you give a name to your function, be careful. Don't use reserved keywords and give reasonable names.
- Two types of function: Functions that return a value(e.g. integer array, double, String etc.) or void type functions(i.e. methods which do not return a value). You must declare it when you define your function.
- Usage(function calling):



Source: Programiz

- Don't manipulate original data (i.e. input parameters) in your functions. You may want to do it for a purpose but if you don't have one, use copy of parameter. Otherwise, if the input parameter is a reference or class type, note that you're changing it elsewhere too.

Objects

- A class is a template that defines the form of an object. It specifies both the data and the code that will operate on that data.
- Syntax:
  - class -class_name- {

    //define member variable(s)

    //define member function(s)

    //define constructor(s) (if you don't define any constructor, default constructor will be utilized.)

  - In main method, you can create objects:

-class_name- -object_name- = new -class_name-();
Example: Person john = new Person();

- Accessing the member functions and member variables:

  * -object_name- . -function_name- (-parameter(s)-) ;

  Example: john.calculateBMI();

  * -object_name- . -variable_name- ; //returns value of member variable.

  Example: john.bmi;

- Java uses a class specification to construct objects. Objects are instances of a class.
- A class is essentially a set of plans that specify how to build an object.
- A class is a logical abstraction. It is not until an object of that class has been created that a physical representation of that class exists in memory.
- Methods and variables that constitute a class are called members of the class.
- The new operator dynamically allocates (that is, allocates at run time) memory for an object and returns a reference to it. This reference is, more or less, the address in memory of the object allocated by new. This reference is then stored in a variable. Thus, in Java, all class objects must be dynamically allocated.