

# OOP

*object oriented programming*

**CLASS** - Has Attributes, Fields, Methods

## CONSTRUCTORS

**Default** (set to zero or null)

**Parameterised** (initialises class variables to parameters passed through it - doesn't have to be all (can be name, age and grade but only put through values for name and age)

## ACCESS MODIFIERS - *control the access*

**Private** = *only accessed within that class*

**Public** = *access from another class*

**Protected** = *can be called by any subclass within the class*

## METHODS - *piece of code within a class to execute something*

**Void** = *does not have a return statement, just executes code*

**Typed** (Data Type) = *want value to be returned. Must have a return statement (can have an array return type - `Public String[].getNames()`)*

**Accessor** - *to get access to class variables* - Typed

**Mutator** - *to change to value of the class variables to the value that is passed through it* - Void

**Auxiliary/Helper** - *add extra functionality to the class* - Void/Typed

**ToString()** - *return values of class as a formatted string* - Typed

**Static** = *do not need to create an object of that class, can call method directly* - **Math Class, Int Class, JOptionPane Class, etc.**

When methods of the class have static in the method heading, it means that it can be called directly instead of creating an object from that class.

**Non-Static** = *have to create an object of that class*

**Method Heading** - *line on top*

**Overloading** - *methods have the same name, but different parameters.*

**Overriding** - *the same named method exists in two different classes.*

**@Override** - do not look at the toString() method in the parent class, use this method to replace this method in the parent class

# Inheritance

*class inherits attributes and methods from another class*

*Parent class is still a child class of main java class.*

*\* Saves RAM and adds to the reusability of code + code from parent class can be used. \**

**Extends** - the class is an extension of another class but still separate

Public class name extends parent class (even without public its still public)

**@override** means that the method of the parent class must be overlooked and the method in the child class must be used.

There is an @override in the parent class because the parent class is a child class of the main Java class.

**Encapsulation** - combination of methods and the properties that the methods act on into one object, as implemented in OOP.

**Information Hiding** - principle of keeping properties and/or methods hidden (as private as possible) from the user (application code) so that the user only has access to code that should be used the application, not all of the code.

**SQL injection** - a website isn't secure so you can insert code to maliciously affect the site. A hacking technique that places code in statements, via web page input.

*Child constructor has parameters for child and parent class*

# More Info From Tests

*Explain the meaning of the “+” and “-” signs and the phrase that is used to describe them. (4)*

Access Modifiers

- indicates that the fields/attributes of the class are private and cannot be accessed outside the class.

+ indicates that the methods of the class are public and can be accessed outside the class.

*Polymorphism (2)*

This refers to the ability of a variable, function or object to take on multiple forms. This allows developers to program in the general rather than program in the specific

*What permanent storage would you use for this particular scenario?*

*Provide TWO reasons for your choice. (3)*

A delimited text file could be used as there is only the data about the Participants to store.

OR

A database could be used with one table to store data for Participants.

Any TWO Reasons:

Easy to backup

Data is portable

*Why would a default constructor be used in the ParticipantViewer class as opposed to a parameterised constructor? (3)*

The default constructor usually would be used to create an array of objects, in this case objects of the Participant class and either gets the information to populate the array from a text file of a database. It usually does not need any data passed to it from the main class.

*A method to search for a participant must be written.*

*12.1 In which class does the programmer write the method?*

*Give a reason for your answer. (3)*

It goes into the ParticipantViewer Class.

This class will contain all the objects that you would must search through to find a participant.

### *Encapsulation (2)*

All data and functions pertaining to an object is held within that object.

### *How could setTariffPoints be used? (2)*

Because this method is private, it can only be called with the Student class. It can be called from the checkTariffPoints method.