

java

Personal notes



[Date]

[Company name]

[Company address]

# Java lesson 1

Public class welcome

ACCESS MODIFIER KEYWORD CLASS NAME

* Sum; variable
* Sum(); function

Function Header

* Return type
* access specifier
* function name
* Parameter list

The main method of the code is strict it can’t be changed.

Once you insert a class and make it the main method the rest of the public function header will appear automatically.

# Addition file

* System.out.prinbt(””)
* System.out.println()
* System.out.printlnf() -Format specifier
* Object,nextInt gets the data from the user. This comes from the Scanner import class

Scanner input = new Scanner(System.in);

* The new keyword creates an object.
* This whole line is called instantiate a Scanner object
* Using the input keyword we can use the methods in the scanner class
* Input is an instance of the class scanner
* Dot operator is how we call the object

# OOP

* Class use to create objects that posses the attributes and behavious defined in the class
* Method defines the program statement that actually performs a task
* A objects attributes define its properties, they are instance variables in class definition
* Instance variables are specific to an object, specified when the object is created
* Classes encapsulate (i.e. wrap) attributes and methods into objects
* Implementation details are hidden within the objects themselves. Information hiding is crucial to good software engineering
* Instantiation: we need to create an object of a class before a program can perform the tasks that the class’ methods define. The process of doing so is called instantiation. An object is referred to as an instance of its class.

## Working with classes

* Object-Oriented Analysis and Design (OOAD) is a method in software development for determining requirements and creating a design that meets them, using an iterative process and object-oriented programming languages like Java.
* Unified Modelling Language (UML) : the most widely used graphical scheme for modelling objectoriented systems
* A static method is special, because we can call it without first creating an object of the class in which the method is declared
* Method invocation(object name followed by dot followed by method name and parameter list
* Class attributes and methods are usually either public or private

## Declaring method with parameter

* A method will need addition information from the calling environment to perform its task in the form of parameters
* Parameters are defined in a comma-separated list enclosed within parentheses following the method name. Each parameter must specify the data type and variable name
* When a method is defined with parameters, the call to the method (i.e. method invocation) must supply an appropriate argument corresponding to each parameter in the method definition
* The number of arguments in a method call must match the number of parameters in the parameter list of the method declaration, and the argument type must be “consistent” with the corresponding parameter type

## Instance variabels, set Methods and get methods

* Local variables are declared within a method and can only be used there. They are lost once the method ends.
* Class attributes are declared as variables in a class declaration, and outside of the bodies of the class’ method declarations.
* When each object of a class maintains its own copy of a class attribute, that attribute is referred to as an instance variable
* Whenever a class has instance variables, it is customary to provide methods that will be used to modify or access the instance variables; these are referred to as set and get methods

## Initializing objects with Constructors

* Java requires each class to have a constructor, a special method that is used to instantiate (i.e. create an instance of) the class
* Keyword new is used when instantiating an object of a class by means of the class’ constructor; this amounts to requesting memory to be allocated for storing the instantiated object
* A constructor has the same name as the class, and has no return type
* By default, the compiler provides a default constructor with no parameters, which can be used to instantiate an object when no explicit constructor has been defined for the class.
* When a default constructor is used, instance variables get their default values. To set custom values, explicitly define a constructor for the class.
* Constructors are normally declared public (since they have to be used by other classes to instantiate objects)

## Class notes

* The class that defines the main method is called the driver class
* Instanve variable must be declared on the top of the class outside
* By declaring a variable inside a class it should be declared using the private access modifier
* For every private variable to access it we use
* Set method: modify and get method: read

# String character and Regular expression

## Literal vs objects

* Literal: an exact representation of a value; usually used for primitives (e.g. int, boolean, character, etc.)
* Character literal: an integer value represented as a character in single quotes
* String literal: a sequence of characters in double quotes, treated as a String object in Java; assigned to a variable via assignment operator (=)

## Class string

* String object: created via the “new” operator
* Java treats all String literals with the same contents as a single String object with many references to it; preserves memory
* Strings in Java are represented as objects
* String Constructors: create a String object, allocating new memory and returning a reference to it in the specified variable
* Observation: String objects in Java are immutable; i.e. once created and initialized, the character contents can’t be changed; thus copying an existing String object is unnecessary (memory misuse)

A screenshot of a computer code

Description automatically generated

* equals returns true if s1, s2 contents are the same; uses lexicographic
* == on String literals means the same as equals; different on String objects created with new operator
* Attempting to instantiate a String literal that already exists in a program does not create a new one
* regionMatches checks for equality of two substrings

## Class notes

* To get the length of a string we will use the the .lenght() method

Int n = color.lenght(); =s

* We do not need an instance of an object inorder to invoke a class method
* Class methods are declared static and is defined for the class and can also be invoked without instantiating a class of the object