CLASSTIME | Page No.
Date 06 / 08 / 18

Experiment - 1

Aim : (a). Factorial of a number
(b). Determine if year is leap year
(c). Fibonacci Series
(d). Palindrome Number

Software Used : Netbeans IDE

Theory

Java is a general-purpose computer-programming language that is concurrent, class-based, object oriented, and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere" (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation.

Java applications are typically compiled to bytecode that can run on any Java Virtual Machine (JVM) regardless of computer architecture.

Java JVM will always look for a specific method signature to start running an application, & that

would be public static void main (String args[]).
Here, "args" is an argument of type String array.
Argument name could be anything, not necessarily args.

```
class Example {
    public static void main (String args[]){
        System.out.print("Hello!");
    }
}
```

Public : It is an Access Modifier, which defines who can access this method. Public means that this method will be accessible by any class.

Static : It is a keyword which identifies the class related thing. It means the given method or variable is not instance related but class related. It can be accessed without creating the instance of the class.

Void : It is used to define the Return Type of the method. Void means the method will not return any value.

main : It is the name of a method. This method name is searched by JVM as a starting point for an application with a particular signature only.

CLASSTIME | Page No.
Date 13 / 08 / 18

Experiment-2

Aim: (a). Write a program to display a greet message according to marks obtained by student.
(b). Calculate Circle Area using radius.
(c). Sum & average of n numbers from user.
(d). Implement stack using stack class.

Software Used: Netbeans IDE

Theory

Reading input from console in Java

In Java, there are three different ways for reading input from the user in the command line environment (console).

1. Using BufferedReader Class
2. Using Scanner Class
3. Using Console Class

The second method, i.e., using Scanner Class is the most preferred method to take input. The main purpose of the Scanner class is to parse primitive types & strings using regular expressions, however it can also be used to read input from the user.

Page No.
Date

Advantages:
- Convenient methods for parsing primitives (nextInt(), nextFloat(), ---) from the tokenized input.
- Regular expressions can be used to find tokens.

Drawback:
- The reading methods are not synchronized.

"Stack" Class in Java

Java collection framework provides a Stack class which models & implements Stack data structure. The class is based on the basic principle of last-in-first-out.

It has two basic methods (among many methods), push() and pop(), used to push & pop an element from the stack respectively.

for e.g-
```
Stack st = new Stack();
st.push(42);
st.push(36);
st.pop();
st.push(29);
st.pop();
```

Page No.
Date 20 08 18

Experiment -3

Aim: (a). Write your own application that creates stack class & extends the class to provide functionality. Use "super" and "this" keyword.
(b). Write a program showing Single level, Multilevel and Hybrid inheritance.

Software Used : Netbeans IDE

Theory

"super" and "this" keywords in Java

"super" keyword is used to access methods of the parent class while "this" is used to access methods of the current class.

Both are reserved keywords in Java, i.e., we can't use them as an identifier. They can be used to refer to class' both instance & static members.

We can use this as well as super anywhere except static area. For example, they can't be used inside public static void main (String args[]) method.

Scope	Access Modifiers			
	private	no modifiers	protected	public
Same class	Yes	Yes	Yes	Yes
Same package subclass	No	Yes	Yes	Yes
Same package non-subclass	No	Yes	No	Yes
Different package subclass	No	No	Yes	Yes
Different package non-subclass	No	No	No	Yes

Page No.
Date

Inheritance in Java

Inheritance is a mechanism in which one object (of a class) acquires all the properties & behaviours of another object (of the parent class). It is an important part of OOP.

The idea behind inheritance is that you can create new classes that are built upon existing ones. When you inherit from an existing class, you can reuse methods & fields of the parent class. Inheritance represents the IS-A relationship which is also known as parent-child relationship.

Syntax of Java Inheritance :

```
class SuperClass {
    // Fields & methods of the SuperClass
}
class SubClass extends Superclass {
    // Fields & methods of the SubClass
}
```

The "extends" keyword indicates that you are making a new class that derives from an existing class. The meaning of "extends" is to increase the functionality.