Page No.
Date

Experiment - 4

Aim : a). Write a program using abstract methods and classes showing Runtime polymorphism.
b). Write a program on Interfaces.

Software Used : Netbeans IDE.

Theory

Abstract Methods and Classes

An abstract class is a class that is declared abstract - it may or may not use abstract methods. Abstract classes cannot be instantiated, but they can be subclassed.

An abstract method is a method that is declared without an implementation (without braces, & followed by a semicolon). If a class includes abstract methods, then the class itself must be declared abstract.

When an abstract class is subclassed, the subclass usually provides implementations for all of the abstract methods in its parent class. However, if it does not, then the subclass must also be declared abstract.

Page No.
Date

Interface

An interface in Java is a blueprint of a class. It is a mechanism to achieve abstraction. There can only be abstract methods in the interface, i.e. functions without body. It is used to achieve abstraction & multiple inheritance in Java.

If a class implements an interface & does not provide method bodies for all functions specified in the interface, then class must be declared abstract.

Page No.
Date 01 | 10 | 18

Experiment - 5

Aim : Exception handling
(a). Write a program using try, catch, throw
(b). Create your own exception class

Software Used : Netbeans IDE.

Theory

An exception is a problem that arises during the execution of a program. When an Exception occurs, the normal flow of the program is disrupted & the program/Application terminates abnormally, which is not recommend, therefore, these exceptions are to be handled.

An exception can occur for many different reasons. Some of these exceptions are caused by user error, others by programmer error, & others by physical resources that have failed in some manner.

Based on these, we have 3 categories of Exceptions. You need to understand them to know how exception handling works in Java.

Page No.
Date

1. Checked Exceptions : A checked exception is an exception that occures at the compile time, these are also called as compile time exceptions. These exceptions cannot simply be ignored at the time of compilation, the programmer should take care of these exceptions. For eg- FileNotFoundException.

2. Unchecked Exceptions : An unchecked exception is an exception that occurs at the time of execution. These are also called as Runtime Exceptions. For eg.- ArrayIndexOutOfBoundsException

3. Errors : These are not exceptions at all, but problems that arise beyond the control of the user or the programmer. Errors are typically ignored in your code because you can rarely do anything about an error. For eg- If a stack overflow occurs, an error will arise. They are also ignored at the time of compilation.