**Exercise 1**

By: Shaheen Ali

There are four fundamental OOP concepts and they are Encapsulation, Inheritance, Polymorphism and Abstraction.

I will now explain all concepts and provide examples.

Encapsulation:

Encapsulation in java is used as to wrap the data or variables and code that is attached to the data giving it meaning as one single unit. In encapsulation, the variables of a class is hidden from other classes and cannot be used or shared and is only local to that class. It can be accessed only through the methods of the current class and this is known as data hiding. It acts like a function with local attribute only belonging to or used in that function.

Example:

/\* File name : EncapTest.java \*/

public class EncapTest {

private String name;

private String idNum;

private int age;

public int getAge() {

return age;

}

public String getName() {

return name;

}

public String getIdNum() {

return idNum;

}

public void setAge( int newAge) {

age = newAge;

}

public void setName(String newName) {

name = newName;

}

public void setIdNum( String newId) {

idNum = newId;

}

}

Inheritance:

In java Inheritance is a class can inherit certain attributes or properties from an existing class. With the use of inheritance, the code can be reused elsewhere and the information is made manageable in a hierarchical order. The class which inherits the properties of other is known as subclass (derived class, child class) and the class whose properties are inherited is known as superclass.

Example:

class Teacher {

String designation = "Teacher";

String collegeName = "Beginnersbook";

void does(){

System.out.println("Teaching");

}

}

public class PhysicsTeacher extends Teacher{

String mainSubject = "Physics";

public static void main(String args[]){

PhysicsTeacher obj = new PhysicsTeacher();

System.out.println(obj.collegeName);

System.out.println(obj.designation);

System.out.println(obj.mainSubject);

obj.does();

}

}

Polymorphism:

Polymorphism is the ability of an object to take on many forms. For example, an animal who has four legs can take on many forms such as of a cat or dog. The most common use of polymorphism in OOP occurs when a parent class reference is used to refer to a child class object.

Example:

class Bike{

void run(){System.out.println("running");}

}

class Splender extends Bike{

void run(){System.out.println("running safely with 60km");}

public static void main(String args[]){

Bike b = new Splender();//upcasting

b.run();

}

}

Abstraction:

Abstraction is the process of abstraction in java is used to hide certain details and only show the essential features of the object. In other words, it deals with the outside view of an object (interface).

Example:

/\* File name : Employee.java \*/

public abstract class Employee {

private String name;

private String address;

private int number;

public Employee(String name, String address, int number) {

System.out.println("Constructing an Employee");

this.name = name;

this.address = address;

this.number = number;

}

public double computePay() {

System.out.println("Inside Employee computePay");

return 0.0;

}

public void mailCheck() {

System.out.println("Mailing a check to " + this.name + " " + this.address);

}

public String toString() {

return name + " " + address + " " + number;

}

public String getName() {

return name;

}

public String getAddress() {

return address;

}

public void setAddress(String newAddress) {

address = newAddress;

}

public int getNumber() {

return number;

}

}