**NAME: NIZEYIMANA Gervais**

**REGNO: 22RP01782**

**CLASS:IT Y2 A**

**Java assignment on oop using java Date:26 January 2024**

**Q1. Object-Oriented Programming** or OOPs refers to languages that use objects in programming. Object-oriented programming aims to implement real-world entities like inheritance, hiding, polymorphism, etc in programming. The main aim of OOP is to bind together the data and the functions that operate on them so that no other part of the code can access this data except that function.

**A brief overview of the main principles of OOP?**

Object-oriented programming is based on the following principles:

* **Encapsulation.**This principle states that all important information is contained inside an object and only select information is exposed. The implementation and state of each object are privately held inside a defined class. Other objects do not have access to this class or the authority to make changes. They are only able to call a list of public functions or methods. This characteristic of data hiding provides greater program security and avoids unintended data corruption.
* **Abstraction.** Objects only reveal internal mechanisms that are relevant for the use of other objects, hiding any unnecessary implementation code. The derived class can have its functionality extended. This concept can help developers more easily make additional changes or additions over time.
* [**Inheritance**](https://www.techtarget.com/whatis/definition/inheritance)**.**Classes can reuse code from other classes. Relationships and subclasses between objects can be assigned, enabling developers to reuse common logic while still maintaining a unique hierarchy. This property of OOP forces a more thorough data analysis, reduces development time and ensures a higher level of accuracy.
* [**Polymorphism**](https://www.techtarget.com/whatis/definition/polymorphism)**.**Objects are designed to share behaviors and they can take on more than one form. The program will determine which meaning or usage is necessary for each execution of that object from a parent class, reducing the need to duplicate code. A child class is then created, which extends the functionality of the parent class. Polymorphism allows different types of objects to pass through the same interface.

## Q2. Difference Between Procedural and Object Oriented Programming

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Procedural Programming** | **Object Oriented Programming** |
| Definition | This programming language makes use of a step by step approach for breaking down a task into a collection of routines (or subroutines) and variables by following a sequence of instructions. It carries out each step systematically in order so that a computer easily gets to understand what to do. | This programming language uses objects and classes for creating models based on the real-world environment. This model makes it very easy for a user to modify as well as maintain the existing code while new objects get created by inheriting the characteristics of the present ones. |
| Security | Procedural Programming does not offer any method of hiding data. Thus, it is less secure when compared to Object Oriented Programming. | Hiding data is possible with Object Oriented Programming due to the abstraction. Thus, it is more secure than the Procedural Programming. |
| Method | The main program gets divided into minute parts on the basis of the functions. It then treats them as separate programs for smaller programs individually. | It involves the concept of classes and objects. Hence, it divides the program into minute chunks known as objects. These are actually instances of classes. |
| Division of Program | Procedural Programming divides the program into small programs and refers to them as functions. | Object Oriented Programming divides the program into small parts and refers to them as objects. |
| Movement of Data | Available data is capable of moving freely within the system from one function to another. | The objects are capable of moving and communicating with each other through the member functions. |
| Approach | The Procedural Programming follows a Top-Down approach. | The Object Oriented Programming follows a Bottom-Up approach. |
| Inheritance | It does not provide any inheritance. | It achieves inheritance in three modes- protected, private, and public. |
| Virtual Classes | There is no concept of virtual classes. | The concept of virtual functions appears at the time of inheritance. |
| Overloading | The case of overloading isn’t possible in the case of Procedural Programming. | Overloading is possible in the form of operator overloading and function overloading in the case of Object Oriented Programming. |
| Reusability of Code | No feature of reusing codes is present in Procedural Programming. | Object Oriented Programming offers the feature to reuse any existing codes in it by utilizing a feature known as inheritance. |
| Basis of World | The Procedural Programming follows an unreal world. | The Object Oriented programming follows the real world. |
| Examples | Some common examples of Procedural Programming are C, Fortran, VB, and Pascal. | The examples of Object Oriented Programming languages are Java, C++, VB.NET, Python, and C#.NET. |

**Q3. Explanation of “class” and “object” in context of java**

**A class** in Java is a set of objects which shares common characteristics/

behavior and common properties/ attributes

 It is a user-defined blueprint or prototype from which objects are created. For example, Student is a class while a particular student named Ravi is an object

**Objects** are the instances of a class that are created to use the attributes and methods of a class.  A typical Java program creates many objects, which as you know, interact by invoking methods.

1. **State**: It is represented by attributes of an object. It also reflects the properties of an object.
2. **Behavior**: It is represented by the methods of an object. It also reflects the response of an object with other objects.
3. **Identity**: It gives a unique name to an object and enables one object to interact with other objects.

**Q4.List and brief description of key andvantages of using oop in java**

1. **Modularity.** Encapsulation enables objects to be self-contained, making troubleshooting and collaborative development easier.
2. **Reusability.** Code can be reused through inheritance, meaning a team does not have to write the same code multiple times.
3. **Productivity.** Programmers can construct new programs quicker through the use of multiple libraries and reusable code.
4. **Easily upgradable and scalable.**Programmers can implement system functionalities independently.
5. **Interface descriptions.** Descriptions of external systems are simple, due to message passing techniques that are used for objects communication.
6. **Security.** Using encapsulation and abstraction, complex code is hidden, software maintenance is easier and [internet protocols](https://www.techtarget.com/searchunifiedcommunications/definition/Internet-Protocol) are protected.
7. **Flexibility.** Polymorphism enables a single function to adapt to the class it is placed in. Different objects can also pass through the same interface.

**Q5.How does oop contribute to code organization and maintenance**

**Object-oriented programming** (OOP) provides a modular and organized approach to software development, enhancing code readability and maintainability. By encapsulating data and behavior within objects, OOP promotes reusability, making it easier to adapt and extend code.

**Q6. What is the java platform, and how does it differ from java programing language**

**The Java platform** is the environment for developing and managing Java applets and applications. It consists of three primary components: the Java language, the Java packages, and the Java virtual machine.

**Java language** is a multi-platform, object-oriented, and network-centric language

​Java platform is a software-only platform that runs on the top of other hardware-based platforms, other platforms are mostly hardware software or hardware only and can be run only on hardware based. Programmer can develop Java code on any OS. Most of the other platforms do not have this capability.

**Q7. How does java achieve independence, and why it is advantageous**

**How does java achieve platform independence**

* Whenever a program is written in JAVA, the javac compiles it.
* The result of the JAVA compiler is the **.class file or the bytecode** and not the machine’s native code (unlike the C compiler).
* The bytecode generated is a non-executable code and needs an interpreter to execute on a machine. This interpreter is the JVM and thus the Bytecode is executed by the JVM.
* And finally, the program runs to give the desired output

**Why is java platform independence advantageous**

Because it can run on multiple platforms