CU CHANDIGARH UNIVERSITY

CHANDIGARH UNIVERSITY

Discover. Learn. Empower.

INSTITUTE : UIE

DEPARTMENT : CSE

Bachelor of Engineering (Computer Science & Engineerin

PROJECT BASED LEARNING IN JAVA

(20CST-319/20ITT-319)

TOPIC OF PRESENTATION:

Polymorphism, Encapsulation and data privacy.

DISCOVER . LEARN . EMPOWE

Lecture Objectives

In this lecture, we will discuss:

Polymorphism, Encapsulation and data privacy.

Polymorphism

Polymorphism in Java is a concept by which we can perform a *single action in different w* Polymorphism is derived from 2 Greek words: poly and morphs. The word "poly" means r and "morphs" means forms. So polymorphism means many forms.

There are two types of polymorphism in Java: compile-time polymorphism and rur polymorphism. We can perform polymorphism in java by method overloading and me overriding.

If you overload a static method in Java, it is the example of compile time polymorphism. I we will focus on runtime polymorphism in java.

There are two types of polymorphism in Java: compile-time polymorphism runtime polymorphism. We can perform polymorphism in java by me overloading and method overriding.

If you overload a static method in Java, it is the example of compile polymorphism. Here, we will focus on runtime polymorphism in java.

Upcasting

If the reference variable of Parent class refers to the object of Child class, it is know upcasting. For example:

class A{}

class B **extends** A{}

A a=**new** B();//upcasting

For upcasting, we can use the reference variable of class type or an interface type. For Example:

interface I{}

class A{}

class B **extends** A **implements** I{}

Here, the relationship of B class would be:

- B IS-A A
- B IS-A I
- B IS-A Object

Since Object is the root class of all classes in Java, so we can write B IS-A Obj

Example of Java Runtime Polymorphism

```
class Bike{
 void run(){System.out.println("running");}
}
class Splendor extends Bike{
 void run(){System.out.println("running safely with 60km");}

 public static void main(String args[]){
  Bike b = new Splendor();//upcasting
  b.run();
 }
}
```

Output:

```
running safely with 60km.
```

Java Runtime Polymorphism with Data Member

Rule: Runtime polymorphism can't be achieved by data members.

Java Runtime Polymorphism with Multilevel Inheritance

Encapsulation

Encapsulation in Java is a *process of wrapping code and data together into a single unit*, for examp capsule which is mixed of several medicines.

- Protective Barrier to prevent data being directly used outside the class
- Hides the implementation level details.

The Java Bean class is the example of a fully encapsulated class.

Advantage of Encapsulation in Java

- make the class **read-only or write-only**.
- It provides you the **control over the data**.
- It is a way to achieve **data hiding** in Java because other class will not be able to access the data through the private data members.
- The encapsulate class is **easy to test**. So, it is better for unit testing.

Data Privacy using Encapsulation

- Fields in a class are made private to prevent it to be accessed by code outside the clas

- Private fields can be accessed only by using the public methods in the class

- It leads to Data Hiding or Privacy

Data Privacy using Encapsulation

- Encapsulated data is accessed using the “Accessor (getter)” and “Mutator (setter)” methods.

- Accessors – Methods to retrieve the hidden data.

- Mutators – Methods to change hidden data.

QUIZ:

1. Which among the following best describes polymorphism?
   a) It is the ability for a message/data to be processed in more than one form
   b) It is the ability for a message/data to be processed in only 1 form
   c) It is the ability for many messages/data to be processed in one way
   d) It is the ability for undefined message/data to be processed in at least one way

2. If same message is passed to objects of several different classes and all of those can respond in a different way, what is this feature called?
   a) Inheritance
   b) Overloading
   c) Polymorphism
   d) Overriding

Summary:

In this session, you were able to :

- Learn about Polymorphism, Encapsulation and data privacy.

References:

Books:

1. Balaguruswamy, *Java.*
2. A Primer, E.Balaguruswamy, *Programming with Java,* Tata McGraw Hill Companies
3. John P. Flynt Thomson, *Java Programming.*

Video Lectures :

https://youtu.be/jg4MpYr1TBc

Reference Links:

https://www.javatpoint.com/runtime-polymorphism-in-java
https://www.javatpoint.com/encapsulation
https://www.tutorialspoint.com/java/java_encapsulation.htm
https://www.geeksforgeeks.org/encapsulation-in-java/

THANK YOU

[redacted]