JAVA – Function overloading

Lecture 7

Polymorphism

"Polymorphism" = "poly" + "morphs"

Many forms

Polymorphism

Allows the object to behave differently in different conditions.

Two types:

1) **Compile time** Polymorphism – Static (or early) binding.
2) **Runtime** Polymorphism – Dynamic (or late) binding.

1) Compile time Polymorphism

- Function overloading
- Operator overloading

2) Run time Polymorphism

- Function overriding

  (using Virtual functions)

Binding:

- Connecting the function call to the function body is called Binding.
- When it is done before the program is run, its called Early Binding or Static Binding or Compile-time Binding.

Function overloading:

Multiple functions with same names but different parameters.

Ways to overload a function:

1. By changing number of Arguments.

```
class Func_Overloading
{
    public:
    int add(int a, int b);
    int add(int a, int b, int c);
    int add(float a, float b);
};
```

Function overloading:

Multiple functions with same names but different parameters.

Ways to overload a function:

1. By changing **number** of Arguments.
2. By having different **types** of argument.

```
class Func_Overloading
{
    public:
    int add(int a, int b);
    int add(int a, int b, int c);
    int add(float a, float b);
};
```

```
public class Test
{
    void m1()
    {
    }
    void m1(int a)
    {
    }
    void m2(int a)
    {
    }
    void m2(float a)
    {
    }
    public static void main(String args[])
    {
    }
}
```

Changing number of arguments

Changing type of arguments

Is Operator Overloading supported by JAVA?

NO

The only operator implicitly overloaded is “+”

FACE

1. A class can have many methods with the same name as long as the number of parameters or type of parameters is different. This OOP concept is known as

a. Method Invocating
b. Method Overriding
c. Method Labelling
d. Method Overloading ✓

2. What is the output of the following program?

```
class Overload {
int x;
int y;
void add(int a)
{
        x = a + 1;
}
void add(int a, int b)
{
        y = b + 2;
}
}
```

```
class Overload_methods {
public static void main(String args[])
{
Overload obj = new Overload();
int a = 0;
int b = 0;
obj.add(6);
obj.add(7,5);
System.out.println(obj.x);
System.out.println(obj.y);
}
}
```

a. 8

3. Find the output for the following code?

```
class A{
public static void main(String args[]){
System.out.println("Hello");
main(5);
}
public static void main(int i){
System.out.println("Hi");
}
}
```

a.Hello b. Hi c. error d. Hello Hi

4. Predict the Output:

```
class A{

public static void method(int i){
System.out.print("Method 1");
}
public static int method(String str){
System.out.print("Method 2");
return 0; } }
public class Test{
public static void main(String args[]){
A obj=new A( );
obj.method(5); } }
```

a. Method 1 b. Method 2

c. Compile time error d. None of the above

5. Which of these can be overloaded?

a. Methods

b. Constructors

c. All of the mentioned ✓

d. None of the mentioned


Thank you