

SNS COLLEGE OF TECHNOLOGY

AN AUTONOMOUS INSTITUTION

COIMBATORE - 35

Department of Computer Science and Engineering

23CSB201 - OBJECT ORIENTED PROGRAMMING

I Year / II Semester

Unit – 2: CONTROL STATEMENTS AND CONSTRUCTORS

Topic – PARAMETERIZED CONSTRUCTORS

Ms. S. Sangeetha, AP-CSD



Parameterized Constructor

Define

A Constructor with arguments(parameters) is known as Parameterized constructor.

Syntax:

```
class ClassName {  
    // Instance variables  
    dataType variableName1;  
    dataType variableName2;  
  
    // Parameterized constructor  
    public ClassName(dataType parameter1, dataType parameter2) {  
        this.variableName1 = parameter1;  
        this.variableName2 = parameter2;  
    }  
}
```



Parameterized Constructor



Ideate

- Initializing objects as the initial state of an object can be easily customized based on the specific values provided at the time of creation.
- The parameterized constructor has to be defined explicitly by the programmer.
- Parameterized constructor can be overloaded.



Parameterized Constructor



Prototype #1

```
public class Car {  
    String model;    int year;  
  
    // Parameterized Constructor  
    Car(String mdl, int yr) {  
        this.model = mdl;  
        this.year = yr;  
    }  
  
    public static void main(String[] args) {  
        Car car = new Car("BA123",2012);  
        System.out.println("Model: " + car.model + ", Year: " + car.year);  
    }  
}
```

Output:
Model: BA123, Year: 2012

Overloading Parameterized Constructor



Prototype #2

```
class Example{
Example(int i, int j){
    System.out.println("parameterized constructor");
    System.out.println("i="+i+", "+"j="+j);
}
Example(int i, int j, int k){
    System.out.println("parameterized constructor");
    System.out.println("i="+i+", "+"j="+j+", "+"k="+k);
}
public static void main(String args[]){
    Example obj1 = new Example(10,20);
    Example obj2 = new Example(10,20,30);
}
}
```

Output:
parameterized constructor
i=10,j=20
parameterized constructor
i=10,j=20,k=30



Parameterized Constructor



Prototype #3

A Weird Compilation Error

```
class Example{  
  
    Example(int i, int j){  
        System.out.println("parameterized constructor");  
        System.out.println("i="+i+", "+"j="+j);  
    }  
  
    public static void main(String args[]){  
        Example obj1 = new Example();  
    }  
}
```

Output:
Error

Explanation

If you do not have any constructor in your class, Java compiler inserts default constructor into your code on your behalf,

however if you implement any constructor then you no longer receive a default constructor.

Calling Default Constructor from Parameterized Constructor

Prototype #4



```
public class Library{
    String studentName;
    int studentRno;
    String member;

    Library(){ member = "YES"; }

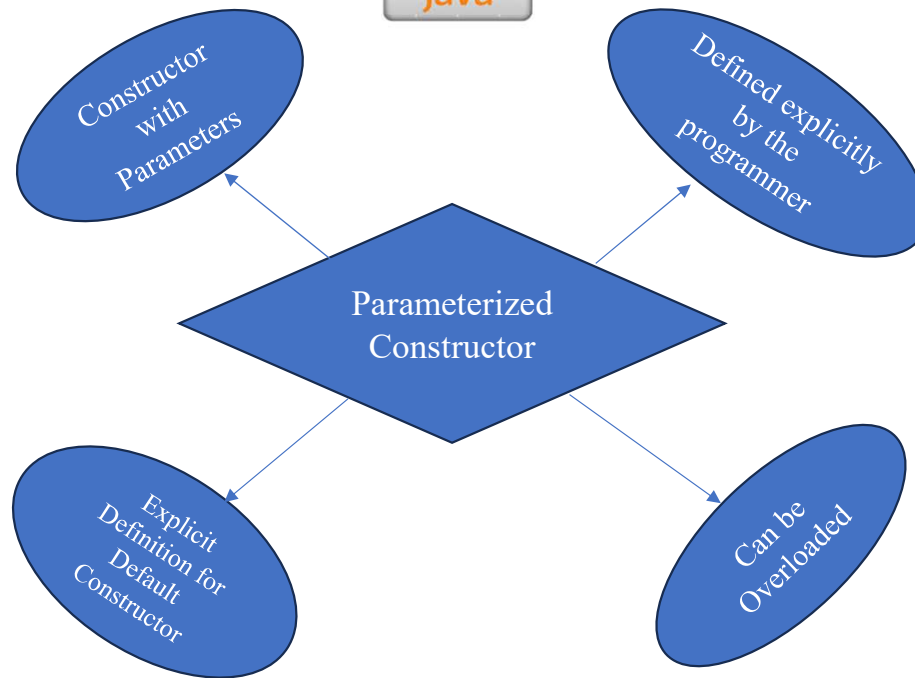
    Library(String name , int rno){
        this();          /* this is used for calling the default constructor*from parameterized constructor*/
        studentName = name;
        studentRno = rno; }

    void display(){
        System.out.println("Name:"+ studentName + " - Rno:" + studentRno+ " - "+ "Member:" + member);}

    public static void main(String args[]){
        Library obj = new Library("Ashok" , 18);
        obj.display();}
}
```

Output:
Name:Ashok - Rno:18 - Member:YES

Mind-map





Puzzles

1. What is the primary purpose of a parameterized constructor?

- A) To return a value after an object is created.
- B) To free up memory when an object is no longer needed.
- C) To initialize the instance variables of a newly created object with user-specified values.
- D) To define the methods a class will contain

Answer: C

2. Can a Java class have both a default constructor and one or more parameterized constructors?

- A) No, a class can only have one type of constructor.
- B) Yes, this is an example of constructor overloading.
- C) Yes, but they must have the same parameter list.
- D) No, this causes a runtime error.

Answer: B

References

1. https://www.w3schools.com/java/java_constructors.asp
2. <https://www.geeksforgeeks.org/java/constructors-in-java/>
3. https://www.tutorialspoint.com/java/java_constructors.htm



