| | |
|---|---|
| Started on | Saturday, 15 February 2025, 8:24 PM |
| State | Finished |
| Completed on | Saturday, 15 February 2025, 8:24 PM |
| Time taken | 49 secs |
| Marks | 5.00/5.00 |
| Grade | **100.00** out of 100.00 |
| Feedback | Congratulations!!! You have passed by securing more than 80% |

## Question 1

1.00/1.00

Observe the below code.

```
public class Student {
  private int id;
  private String name;
  private char grade;
  //Constructor 1
 public Student() {
    id=0;
    name= " ";
  }
  //Constructor 2
 public Student(int id, String name) {
    this.id=id;
    this.name=name;
 }
}
```

Choose the constructor that is invoked, when an object is created as shown below.

new Student();

Constructor 1

✔

new Student(54, "John");

Constructor 2

✔

Your answer is correct.

The correct answer is:

Observe the below code.

```
public class Student {
  private int id;
  private String name;
  private char grade;
  //Constructor 1
 public Student() {
    id=0;
    name= " ";
  }
  //Constructor 2
 public Student(int id, String name) {
    this.id=id;
    this.name=name;
 }
}
```

Choose the constructor that is invoked, when an object is created as shown below.

new Student(); [Constructor 1]

new Student(54, "John"); [Constructor 2]

## Question 2

1.00/1.00

You are given with few classes.

Match the code with the type of constructor available in the class.

```
public class Student {
}
```

Default constructor ✔

```
public class Student {
      public Student() {
      }
}
```

No-argument Constructor ✔

```
public class Student {
      public Student(int studentId, String name) {
      }
}
```

Parameterized Constructor ✔

Your answer is correct

Your answer is correct.

The correct answer is: public class Student {
} → Default constructor, public class Student {
    public Student() {

    }

} → No-argument Constructor, public class Student {
    public Student(int studentId, String name) {

    }

} → Parameterized Constructor

## Question 3

1.00/1.00

Observe the code below.

```
public class Student {
        int studentId;
        String name;
        char grade;

    public Student(int studentId, String name, float mark) {
            this.studentId = studentId;
            this.name=name;
            calculateGrade(mark);
    }

    public void calculateGrade(float mark){
        if(mark>90)
            grade='A';
        else
            grade='B';
    }
}
```

For the code

```
Student s = new Student(1,"Peter",95);
```

What will be the output?

Select one:

- Compilation error because of the parameter – mark - in constructor. It should be grade instead of mark.
- Compiles successfully ✔
- Compilation error because cannot call methods from constructor

Your answer is correct.

The correct answer is: Compiles successfully

## Question 4

1.00/1.00

Observe the below class.

```
class Product{
  int productId;
  String productName;
  Product    {
       productId=0; productName=" ";
  }
  Product(int id, String name)  {
       //access Product()  ---- Line 1
      productId=id;
      productName=name;
  }
}
```

Identify the valid option which is used to invoke the no argument constructor, Product(), at Line 1.

Select one:

- Product();
- Product
- super();
- this();✔

this() invokes the current object's no argument constructor.

The correct answer is: this();

## Question 5

1.00/1.00

choose the code so that the constructor for Student class is overloaded correctly.

```
public class Student {
private int studentId;
private String name;
private float grade;
public Student(int studentId, String name) {
```

this.studentId=studentId;this.name=name;

}

public Student(int studentId, String name, float grade) {

this(studentId,name);

this.grade=grade;

}

}

Your answer is correct.

The correct answer is:

choose the code so that the constructor for Student class is overloaded correctly.

public class Student {

private int studentId;

private String name;

private float grade;

public Student(int studentId, String name) {

[this.studentId=studentId;this.name=name;]

}

public Student(int studentId, String name, float grade) {

[this(studentId,name);]

[this.grade=grade;]

}

}