Started on	Saturday, 27 April 2024, 12:28 PM
State	Finished
Completed on	Saturday, 27 April 2024, 1:03 PM
Time taken	35 mins 2 secs
Marks	15.00/30.00
Grade	**50.00** out of 100.00

Question **1**

Complete

Mark 0.00 out of 1.00

Consider the following code inside Bank.java file:

```
class Bank {
        String bankName, area, phoneNo;
        bank(){
                System.out.println("Banking application");
        }
        public static void main(String[] args) {
                Bank bank = new Bank();
                System.out.println(bank.bankName);
        }
}
```

What happens, when the program executes?

Select one:

- **a. Print null**
- **b. Print “Banking application ” null**
- **c. Compile time error as the bank() method doesn’t have return type specified**
- **d. Compile time error as the constructor name and class name are not same**

Question **2**

Complete

Mark 0.00 out of 1.00

Consider the following code inside the Demo.java file:

```
public class Demo{
        private static void main(String[] args){
                System.out.println("Hello World");
        }
}
```

What happens, when the program executes?

Select one:

- **a. No output will be generated**
- **b. Runtime exception**
- **c. Display “Hello World”**
- **d. Compilation error**

Question 3

Complete

Mark 1.00 out of 1.00

What will be the output when we execute the following code snippet?

```
public class Bank{
    String bankName;
    String area;
    String phoneNo;
    public static void main(String[] args){
        Bank bank2=new Bank("abc","xyz","pqr");
        System.out.println(bank2.bankName);
        System.out.println(bank2.area);
    }
    Bank(String bankName,String area,String phoneNo){
        bankName=bankName;
        this.area=area;
    }
}
```

Select one:

- **a. abc xyz**
- **b. null null**
- **c. abc null**
- **d. null xyz**

Question 4

Complete

Mark 0.00 out of 1.00

Consider the following code inside Account.java file:

```
class Account {
        double balance;
        public static void main(String[] args) {
                Account account1=null; //Line1
                Account account2=null; //Line2
                account1 = new Account(); // Line 3
                account2 = new Account(); //Line 4
                account2 = account1; //Line 5
                account1 = new Account(); // Line 6
        System.out.println(account1.balance);

        }
}
```

Which of the following statement is true after executing line 6?

Select one:

- **a. account1 and account2 are pointing to the same object**
- **b. account1 is pointing to two different object at the same time**
- **c. account2 is pointing to two different object at the same time**
- **d. account1 and account2 are pointing to the different object**

Question 5

Complete

Mark 1.00 out of 1.00

```
class JavaApplication {
        public static void main(String[] args){
                boolean a= true;
                boolean b=!true;
                boolean c= a|b;
                boolean d=a&b;
                boolean e= d?b:c;
                System.out.println(d+""+e);
        }
}
```

What happens, when the program executes?

Select one:

- **a. true true**
- **b. true false**
- **c. false**
- **d. false true**

Question 6

Complete

Mark 0.00 out of 1.00

Bank application starts with a login screen. If the username=”John” and password="Infy123", We need to display the message as “Welcome!” else we need to display "Sorry, wrong credentials!". What is the right way to do it?

Select one:

- **a. both option 1 and option 2**
- **b. None of the option**
- **c. if (username. equals("John") && password. equals("Infy123")) System.out.println ("Welcome!"); else System.out.println ("Sorry, wrong credentials!");**
- **d. String s = (username. equals (”John”) && password. equals("Infy123"))? "Welcome!”: "Sorry, wrong credentials!”; System.out.println(s);**

Question 7

Complete

Mark 1.00 out of 1.00

Given:

```
int i = 1,j = 10;
                do {
                        if(i>j) {
                                break;
                         }
                         j--;
                } while (++i <5);
                System.out.println("i =" +i+" and j = "+j);
```

What is the result?

Select one:

- a. i = 5 and j = 5
- b. i = 5 and j = 6
- c. i = 6 and j = 4
- d. i = 6 and j = 5

Question 8

Complete

Mark 1.00 out of 1.00

Consider the following code inside Bank.java file:

```
class Bank {
        String bankName, area, phoneNo;

        public static void main(String[] args) {
                Bank bank = new Bank();
                System.out.println(bank.bankName);
        }
}
```

Select one:

- a. Runtime exception as none of the variables are initialized
- b. null
- c. Executes successfully with no output
- d. Compile time error as the constructor is not provided

Question 9

Complete

Mark 0.00 out of 1.00

What will be the output of the below code snippet?

```
class Loan{
        Loan(){
                System.out.println("Request for loan");
        }
}
class HomeLoan extends Loan{
        public static void main(String[] args){
                HomeLoan obh=new HomeLoan();
        }
}
```

Select one:

- **a. Executes successfully, with no output displayed**
- **b. Runtime exception**
- **c. compilation error as child class also should have a constructor, with super() in it**
- **d. Request for loan**

Question 10

Complete

Mark 0.00 out of 1.00

What would be the output when the following code is compiled/executed?

```
class Demo{
        Demo (int i){
                System.out.println(i);
                this.i = i;
        }
        public static void main(String[] args){
                Demo obj = new Demo(10);
        }
}
```

Select one:

- **a. 0**
- **b. Compile error "No variable i defined in class ABC**
- **c. 10**
- **d. null**

Question 11

Complete

Mark 1.00 out of 1.00

Consider the following code inside Demo.java file

```
class Demo{
        public static void main(String[] args) {
        int x=200;
        int y=x;
        x++;
        int z=y;
        z++;
        System.out.println(y);
        }
}
```

What happens, when the program executes?

Select one:

- **a. 200**
- **b. 101**
- **c. 0**
- **d. 102**

Question 12

Complete

Mark 1.00 out of 1.00

Consider the following code inside Bank.java file:

```
class Bank{
        String bankName,area,phoneNo;
        public static void main(String[] args){
                Bank bank=new Bank();
                System.out.println(bank.bankName);
                Bank bank2=new Bank("abc","xyz","pqr");
                System.out.println(bank2.bankName);
        }
        Bank(){
                 System.out.println("inside constructor");
        }
        Bank(String x,String y,String z){
                 bankName=x;
                 area=y;
                 phoneNo=z;
        }
}
```

What will be the output when we run the code?

Select one:

- **a. inside constructor null abc**
- **b. Runtime exception as none of the variables are initialized**
- **c. print “abc” but with Runtime exception**
- **d. null abc**

Question **13**

Complete

Mark 0.00 out of 1.00

What is the output of the following code snippet?

```
class Person{
        String name;
        Person(){
       System.out.println("In Person class");
    }
}
class Student extends Person{
        Student(){
      System.out.println("In Student class");
    }
        int id;
        public static void main(String[] a){
                Student ob=new Student(); //Line 1
                ob.name="abc"; //Line 2
                ob.id=123;}
}
```

Select one:

- **a. In Student class In Person class**
- **b. Compilation error in Line 2**
- **c. In Student class**
- **d. In Person class In Student class**

Question **14**

Complete

Mark 0.00 out of 1.00

Which of the following are valid array declarations/definitions?

```
1: int[5] myArray1;
2: int[] myArray2;
3: int[] myArray3 = new int[5];
4: int[5] myArray4 = new int[5];
5: int []myArray5 = new int[5];
6: int[] myArray6 = new int[];
7: int[] myArray7 = null;
```

Select one:

- **a. 3,4,5,7**
- **b. 2,3,5,7**
- **c. 2,4,5,7**
- **d. 1,2,3,6**

Question 15

Complete

Mark 0.00 out of 1.00

Consider the following class inside DelhiOffice.java file

```
abstract class Bank {
        private String bankName;
   Bank(String bankName) {
                this.bankName = bankName;
          }
       public String getBankName() {
                return bankName;
       }
}
class DelhiOffice extends Bank {
           DelhiOffice() {
                   super("Axis Bank");
            }
            public static void main(String[] args) {
                   Bank bank = new DelhiOffice();
                   System.out.println(bank.getBankName());
            }
}
```

What will happen after compilation of the code?

Select one:

- **a. Compilation error will occur because "abstract class must have an abstract method"**
- **b. Compilation error will occur while invoking the super class constructor**
- **c. Code will be compiled successfully**
- **d. Compilation error will occur because "abstract class cannot have constructor"**

Question 16

Complete

Mark 1.00 out of 1.00

Consider the following code inside Bank.java file:

```
class Bank{
           String name;
           int createAccount(int accountNo,String accountHolderName){
                   String name;
                   int i=10;
                   name= accountHolderName; // Line 5
                   return i;
           }
}
```

In method createAccount at Line 5. What is the scope of the variable "name"?

Select one:

- **a. instance variable**
- **b. local variable**
- **c. class variable**
- **d. static variable**

Question 17

Complete

Mark 1.00 out of 1.00

Consider the following code inside Bank.java file:

```
class Bank{
        String bankName;
        Bank(){
                bankName="TrustBank";
        }
}
```

Choose the correct option for the above code.

Select one:

- **a. Not a valid constructor. As it is having same name as class**
- **b. Code snippet is fine**
- **c. Not a valid constructor declaration. As constructor should take some arguments**
- **d. Not a valid Constructor declaration. As constructor is also a method it should have return type associated with it**

Question 18

Complete

Mark 1.00 out of 1.00

What will be the output of the following code snippet?

```
class Loan{
        int i;
        Loan(int x){
                i=x;
                System.out.println("Request for loan");}
}
class HomeLoan extends Loan{
        int i;
        HomeLoan(){
          super(10);
                i=super.i;
                System.out.println("Request for homeloan"+i);
    }
    public static void main(String[] args){
                HomeLoan obh=new HomeLoan();
    }
}
```

Select one:

- **a. compilation error due to super.i statement**
- **b. executes successfully, with no output displayed**
- **c. Request for loan Request for homeloan 10**
- **d. Runtime exception**

Question **19**

Complete

Mark 1.00 out of 1.00

John is writing a class to handle the Account holder details. It requires two important variables as name and mobileNumber. For name, he has taken the datatype as String. For mobileNumber what should be the datatype?

Select one:

- **a. float**
- **b. char**
- **c. int**
- **d. long**

Question **20**

Complete

Mark 1.00 out of 1.00

Consider the following code inside Demo.java file

```
class Demo{
        public static void main(String[] args){
                int m = 1;
        int n = ++m + m++ + --m;
        System.out.println(n);
        }
}
```

What happens, when the program executes?

Select one:

- **a. 6**
- **b. 7**
- **c. 8**
- **d. 5**

Question 21

Complete

Mark 0.00 out of 1.00

What will be the output of the following code snippet?

```
class Loan{
        Loan(int x){
                System.out.println("Request for loan");
        }
}
class HomeLoan extends Loan{
        HomeLoan(){
                System.out.println("Request for HomeLoan");
        }
        public static void main(String[] args){
                HomeLoan obh=new HomeLoan();
        }
}
```

Select one:

- **a. Request for loan**
- **b. Executes successfully, with no output displayed**
- **c. Runtime exception**
- **d. Compilation error: Implicit super constructor Loan() is undefined. Another constructor must be invoked explicitly.**

Question 22

Complete

Mark 0.00 out of 1.00

Consider the following code inside ABCBank.java file

```
class ABCBank{
        int phone;
        double getPhone(){
                System.out.println("phone number received");
                return phone;
        }
}
```

What happens, when the program executes?

Select one:

- **a. The code will compile successfully and generate ABCBank.java file**
- **b. The code will not compile as the return type in the getPhone method should be int not double**
- **c. The code will not compile as there is no main method**
- **d. The code will compile successfully and generate ABCBank.class file**

Question **23**

Complete

Mark 1.00 out of 1.00

Which of the following is true regarding JVM?

Select one:

- **a. None of the above**
- **b. JVM is platform-independent**
- **c. JVM compiles the bytecode, converts it into native code, and then executes the native code**
- **d. JVM directly passes the bytecode to the processor, and as a result the code is executed**

Question **24**

Complete

Mark 1.00 out of 1.00

What is the output for the following program?

```
class ArrayProgram {
        public static void main(String[] args)
            {
                int myArr[] = new int[] {0 , 1, 2, 3, 4, 5, 6, 7, 8, 9};
                int n = 6;
                n = myArr[myArr[n] / 2];
                System.out.println(myArr[n] / 2);
            }
}
```

Select one:

- **a. 1**
- **b. 0**
- **c. 6**
- **d. 3**

Question **25**

Complete

Mark 0.00 out of 1.00

John and his teammate Paul are working on a Java project. John is working on a Windows machine whereas Paul is working on a Mac machine.
Which feature of Java helps John and Paul's projects to execute on each other's machines, even though they are working in different environments?

Select one:

- **a. Object oriented**
- **b. Architecture neutral and portable**
- **c. Multithreading**
- **d. Memory management**

Question **26**

Complete

Mark 0.00 out of 1.00

What will be the output of the following code?

```
class Demo{
        int i;
        Demo(int i){
                System.out.println(i);
                this.i = i;
        }
        public static void main(String[] args){
                Demo obj = new Demo(10);
                System.out.println(obj.i);
        }
}
```

Select one:

- **a. 0 10**
- **b. 10 0**
- **c. 10 10**
- **d. Compilation error, instance variable “i” is not initialized**

Question **27**

Complete

Mark 0.00 out of 1.00

Consider the following code:

```
class Demo{
        public static void main(String[] args){
                int i = 1,j = -1;
                switch (i) {
                case 0, 1:
                        j = 1;
                case 2:
                        j = 2;
                default:
                        j = 0;
                }
                System.out.println("j="+j);
        }
}
```

What will happen on executing the code?

Select one:

- **a. j = -1**
- **b. j = 0**
- **c. Compilation fails**
- **d. j = 1**

Question **28**

Complete

Mark 1.00 out of 1.00

What is the output of the following code snippet?

```
class Person{
        String name;
        Person(){
       System.out.println("In Person class");
    }
}
class Student extends Person{
        Student(){
       System.out.println("In Student class");
   }
        int id;
        public static void main(String[] a){
                Student ob=new Student(); //Line 1
                ob.name="abc"; //Line 2
                ob.id=123;}
}
```

Select one:

- **a. In Person class In Student class**
- **b. In Student class In Person class**
- **c. In Student class**
- **d. Compilation error in Line 2**

Question **29**

Complete

Mark 1.00 out of 1.00

Which of the following is proper declaration of a final method in Java?

Select one:

- **a. public final void method(){ }**
- **b. final public void method();**
- **c. public void final method() { }**
- **d. public final method();**

Question **30**

Complete

Mark 0.00 out of 1.00

Consider the following code inside Account.java file

```
class Person{
        String name;
        Person(){
                name="John";
        }
}
class Employee extends Person{
         int age;
         Employee(){
                  age=34;
          }
}
class Customer extends Person{
         int salary;
         Customer(int salary){
                   this.salary=salary;
              name="Maddy";
         }
                  public void displayDetails(){
                  System.out.println(name+age+salary);
         }
}
class Account {
    public static void main(String[] args) {
                Customer c=new Customer(20000);
                c.displayDetails();
        }
}
```

What will happen when the code is executed?

Select one:

- **a. Code executes successfully and display “John3420000”**
- **b. Compilation fail “the variable age is not accessible from displayDetails method ”**
- **c. Code executes successfully and display “Maddy3420000”**
- **d. Compilation fail “Person cannot have two child classes i.e. Employee and Customer”**