|  |  |
| --- | --- |
| **1. What is the result of compiling and running the following code?**  **class Base {**  **protected void foo() {}**  **}**  **class Derived extends Base {**  **void foo() {}**  **}**  **public class Main {**  **public static void main(String args[]) {**  **Derived d = new Derived();**  **d.foo();**  **}**  **}**  **Output: Compilation error** | **2 Which of the following is used with the switchstatement?** **a)Continue** **b)Exit** **c)break** **d)do** |
| **3)**  **class Mud {**  **12. // insert code here**  **13. System.out.println("hi");**  **14. }**  **15. }**  **And the following five fragments:**  **public static void main(String...a) {**  **public static void main(String.\* a) {**  **public static void main(String... a) {**  **public static void main(String[]... a) {**  **public static void main(String...[] a) {**  **How many of the code fragments, inserted**  **independently at line 12, compile?** | **4). Which of the following is not a type of inheritance ?**  **a.Multiple**  **b.Multilevel**  **c.Distributive**  **d.Hierarchial** |
| **5) Which of the following concepts means wrapping up of data and functions together?**  **A. Abstraction**  **B. Encapsulation**  **C. Inheritance**  **D. Polymorphism** | **6) what is the output for the below code?**  **public class Test{**  **public static void main(String args[]){**  **int i1 =1;**  **switch(i1)**  **case 1:System.out.println("One");**  **case 2:System.out.println("Two");**  **case 3:System.out.println("Three");**    **}**  **}**  **a.OneTwoThree**  **b.One**  **c.OneTwo**  **d.Compile error** |
| **7. . Which type of class allows only one object of it to be created.**  **A. Virtual Class**  **B. Abstract Class**  **C. Singleton Class**  **D. Friend Class** | **8. What is the output of this program?**  **class operators {**  **public static void main(String args[]){**  **int var1 = 5;**  **int var2 = 6;**  **int var3;**  **var3 = ++ var2 \* var1 / var2 + var2; System.out.print(var3);**  **}**  **}**  **a) 10 b) 11 c) 12 d) 56** |
| **9. Predict the output of the following code**  **class Test {**  **int x;**  **int y;**  **void add(int a) {**  **x = a + 1;**  **}**  **void add(int a, int b) {**  **x = a + 2;**  **}**  **public static void main(String[] args) {**  **Test obj = new Test();**  **int a = 0;**  **obj.add(6);**  **System.out.println(obj.x);**  **}}**  **Output: 7** | **10.** **Predict the output of the following code**  **public class Tester {**  **public static void main(String[] args) {**  **String a = "rooman";**  **String b = "rooman";**  **String c = new String("rooman");**  **System.out.print(a==b);**  **System.out.print(a==c);**  **System.out.print(b.equals(c));**  **System.out.print(b.equals(a));**  **}**  **}**  **Output: True,False,True,True** |
| **11. What is the result of compiling and running the following program?**  **class ExamDemo2{**  **static Byte b;**  **public static void main(String [] args){**  **b=10;**  **b+=10;**  **System.out.println(b);**  **}**  **}**  **1 Compilation Fails**  **2 20**  **3 Run Time Error** | **12. What is the output of the following program?**  **class MyClass {**  **public static void main(String[] args) {**  **System.out.println("www." + test() + ".com");**  **}**  **static void test() {**  **System.out.println("rooman");**  **}}**  **Output: Compilation Error** |
| **13. Predict the output**  **public class Calculator {**  **int num = 100;**  **public void calc(int num) {**  **this.num = num \* 10;**  **}**  **public void printNum(){**  **System.out.println(num);**  **}**  **public static void main(String[] args) {**  **Calculator obj = new Calculator ();**  **obj.calc(2);**  **obj.printNum();**  **}**  **}**  **What is the result?**  **A) 20 B) 100**  **C) 1000 D) 2** | **14. What is the result of compiling and running the following code?**  **public class MyLoop {**  **public static void main(String[] args) {**  **String[] sa = {"tom ", "jerry "};**  **for (int x = 0; x < 3; x++) {**  **for (String s : sa) {**  **System.out.print(x + " " + s);**  **if (x == 1) {**  **break;**  **} } } }}**  **A) 0 tom 0 jerry 1 tom 1 jerry**  **B) 0 tom 0 jerry 2 tom 2 jerry**  **C) 0 tom 0 jerry 1 tom 2 tom 2 jerry**  **D) 0 tom 0 jerry 1 tom 1 jerry 2 tom 2 jerry** |

**16) What is overloading and overriding in java?**

**Overloading:** The process of specifying multiple methods, having different signature with same method name. In method overloading, the methods must have same name, having either different no of parameters or different type of parameters or different order of parameters. In this the compiler will decide which method has to be executed and it is called compile time polymorphism.

**Overriding:** The process of specifying two methods with same signature and same return type in two different classes that have IS-A relationship. In this when a method is invoked, the decision of which method to be executed is taken by the JVM at run time and this is called run time polymorphism.

**17) Can we have multiple public classes in a java source file??**

**Ans:** No, while defining multiple classes in a single Java file you need to make sure that only one class among them is public. If you have more than one public classes in a single file a compile time error will be generated.

**18)Write difference between static and non-static variable and methods?**

**Static Variables:**

1.A static variable can be accessed by static members as well as non-static member functions.

2, A static variable acts as a global variable and is shared among all the objects of the class.

3. Static variables occupies less space and memory allocation happens once.

4. A static variable is declared using static keyword.

**Non-static variables:**

* Anon static variable can not be accessed by static member functions.
* A non static variables are specific to instance object in which they are created.
* A non static variable may occupy more space.Memory allocation happen at run time.
* A Normal variable is not required to have any special keyword.

**Static Methods:**

1.A static method can access only static members ans can not access non static members.

2.Static method uses compile time binding or early binding.

3.A static method cannot be overridden being compile time binding.

4.Static method occupies less space and memory allocation happens once.

5.a static method is declared using static keyword.

**Non-Static Methods:**

1.A non static method can access both static as well as non static members.

2.A non static method can be overridden being dynamic binding.

3.A non static method occupy more space. Memory allocation happens when method is

invoked and memory is deallocated once method is executed completely.

4.A normal method is not required to have any special keyword.

**19) What is final key word in java?.**

**Ans:** The final keyword is a non-access modifier used for classes ,attributes and methods, Which makes them non-changeable(impossible to inherit or override).The final keyword is useful when you want a variable to always store the same value.

**20) What is Inheritance and explain about types of Inheritances in java?**

**Inheritance:** It is a mechanism of acquiring the features and behaviours of a class by another class. The class whose members are inherited is called the base class, and the class the inherits those members is called the derived class.

**OOPs support the six different types of inheritances.**

* **Single Inheritance:** In this Inheritance a derived class is created from a single base class
* **Multi-level Inheritance:** In this Inheritance a derived class is created from another derived class.
* **Multiple Inheritance:** In this Inheritance, a derived class is created from more than one base class.
* **Hierarchical Inheritance:** In this Inheritance, more than one derived classes are created from a single base class and further child classes act as parent classes for more than one child classes.
* **Hybrid Inheritance:** This is a combination of more than one Inheritance. It may be a combination of multi-level and multiple or Hierarchical and multilevel or Hierarchical and multipath or Hierarchical and multiple and multilevel.

**21)Write a program to check whether the given name is palindrome or not?**

public class IsPalindrome {

public static void main(String[] args){

String rev="";

Scanner sc = new Scanner(System.in);

System.out.println("Enter any string");

String str=sc.nextLine();

int len =str.length();

for(int i=len-1;i>=0;i--) {

rev = rev + str.charAt(i);

}

if (str.equals(rev)) {

System.out.println("Given string is palindrome");

}

else

System.out.println("Given String is not palindrome");

}

}

**22)Write a program display quadratic sequence to the given number**

**input :5**

**1 3 6 10 15**

**23)** **Write a program display odd palindrome number in the given range.**

public class PalindromeRange {

static boolean checkPalindrome(int number) {

boolean b = false;

int rem =0, rev=0 ,temp=number;

while(number > 0) {

rem = number % 10;

rev = rev \* 10 + rem;

number = number / 10;

}

if(temp == rev) {

b = true;

}

return b;

}

static boolean isOdd(int number) {

boolean b = false;

if(number % 2 != 0) {

b = true;

}

return b;

}

static String rangePalindrome(int startValue, int endValue) {

String res = "";

for (int i = startValue;i <= endValue;i++) {

if(checkPalindrome(i) && isOdd(i)) {

res += i + " ";

}

}

return res;

}

public static void main(String[] args) {

System.out.println(rangePalindrome(100,1000));

}

}

**24)Write a program accept emailid, return the masked email id.**

**Input :testmail@gmail.com**

[**Output:teXXXXXX@gmail.com**](mailto:Output:teXXXXXX@gmail.com)

public class MaskedEmailId {

static String maskMailId(String email) {

String res = "";

res += email.replace("stmail", "XXXXXX");

return res;

}

public static void main(String[] args) {

String email = "testmail@gmail.com";

System.out.println(maskMailId(email));

}

}

**25.Write a program find factorial of given number using recursive**

**function**

public class FactorialUsingRecursion {

static int factorial(int n) {

if (n==0) {

return 1;

}

else {

return(n \*factorial(n-1));

}

}

public static void main(String[] args) {

int i,fact=1;

int num = 5;

fact = factorial(num);

System.out.println("Factorial of " + num+ " is : " + fact);

}

}