PROGRAMMING IN JAVA

Assignment 0

TYPE OF QUESTION: MCQ

Number of questions: 10   Total marks**: 10 × 1 = 10**

QUESTION 1:

What will be the output of the following code snippet?

```
#include <stdio.h>
void solve() {
    int a = 3;
    int res = a++ + ++a + a++ + ++a;
    printf("%d", res);
}
int main() {
    solve();
    return 0;
}
```

a. 12
b. 24
c. 20
d. 18

Correct Answer: c. 20

Detailed Solution:

In the given code snippet, the `solve()` function is called from the `main()` function. Inside the `solve()` function: `int a = 3;` initializes the variable `a` with the value 3. `int res = a++ + ++a + a++ + ++a;` performs a series of arithmetic operations and assigns the result to the variable res. `a++` is a post-increment operation, which means the value of a is used and then incremented by 1. So, initially, `a++` evaluates to 3, and a becomes 4. `++a` is a pre-increment operation, which means the value of `a` is incremented by 1 and then used. So, after the previous operation, `++a` evaluates to 5, and `a` becomes 5. `a++` is another post-increment operation. At this point, `a` is 5, and the expression evaluates to 5. Then, `a` becomes 6. `++a` is another pre-increment operation. After the previous operation, `++a` evaluates to 7, and `a` becomes 7. Therefore, the overall expression becomes 3 + 5 + 5 + 7, which results in 20 and is stored in the variable `res`. `printf("%d", res);` prints the value of `res`, which is 20.

QUESTION 2:

What will be the value of x in the following code snippet?

```
#include <stdio.h>
void solve() {
    int x = printf("Hello");
    printf(" %d", x);
}
int main() {
    solve();
    return 0;
}
```

a. 10
b. 5
c. 1
d. 0

Correct Answer: b. 5

Detailed Solution:

`printf` is a library function defined under `stdio.h` header file that returns the count of characters printed in `STDOUT`. In this case it prints *"Hello"*, which is 5 characters, thus the value returned to x is 5.

QUESTION 3:
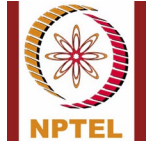
Pick the best option for the following statement:

```
int (*p)[5];
```

a. The statement will create an array `p` of pointers.
b. he statement has an error.
c. The statement will create an array `p` of integers.
d. None of the above

Correct Answer: a. The statement will create an array p of pointers.

Detailed Solution:

The statement does not have an error and will compile normally. `"int (*p)[5];"` is same as `"int* p[5];"` which is used to declare an array of pointers.
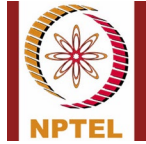
QUESTION 4:

The size of a union is determined by:

  a. The sum of all the members' sizes
  b. The biggest member of the union
  c. The last member of the union
  d. The first member of the union

Correct Answer: b. The biggest member of the union

Detailed Solution:

In a union, all members share the same memory space, and the size of the union is determined by the size of its largest member. This is because the memory allocated for a union needs to be large enough to accommodate the largest member so that it can hold any of the union's members. When calculating the size of a union, it is sufficient to consider only the largest member. The sizes of other members do not contribute to the overall size of the union since they share the same memory space.

QUESTION 5:

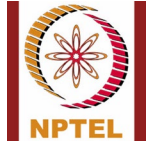**What will be the output of the following code snippet?**

```c
#include <stdio.h>
int main()
{
        int *ptr, a = 10;
        ptr = &a;
        *ptr += 1;
        printf("%d,%d", *ptr, a);
}
```

a. 10, 10
b. 10, 11
c. 11, 10
d. 11, 11

Correct Answer: d. 11, 11

Detailed Solution:

Since `ptr` is just a pointer to `a`, when the value of a changes, so does the value in the pointer. Thus `*ptr += 1;` will increment the value of `a` as well.

QUESTION 6:

What will be the output of the following code snippet?

```
#include <stdio.h>
// Assume base address of "Test_Quiz" to be 1000
int main()
{
   printf(5 + "Test_Quiz");
   return 0;
}
```

a. Test
b. Quiz
c. 1005
d. Compile-time error

Correct Answer: b. Quiz

Detailed Solution:

`printf` is a library function defined under `stdio.h` header file. The compiler adds `5` to the base address of the string through the expression `5 + \"Test_Quiz\"` . Then the string `\"Quiz\"` gets passed to the standard library function as an argument. Then the string is printed normally without error.

QUESTION 7:

Which of the following is not a storage class specifier in C?

    a. auto
    b. register
    c. static
    d. volatile

Correct Answer: d. volatile

Detailed Solution:

`volatile` is not a storage class specifier in C. `volatile` and `const` are type qualifiers.

QUESTION 8:

In C, static storage class cannot be used with:

a. Global variable
b. Function parameter
c. Function name
d. Local variable

Correct Answer: b. Function parameter

Detailed Solution:

Declaring a global variable as static limits its scope to the same file in which it is defined. A static function is only accessible to the same file in which it is defined. A local variable declared as static preserves the value of the variable between the function calls.

QUESTION 9:

What will be the output of the following code snippet?

```
#include<stdio.h>
int main()
{
    char str[20] = "Test_Quiz";
    printf ("%d", sizeof(str));
    return 0;
}
```

a. 9
b. 10
c. Garbage value
d. 20

Correct Answer: d. 20

Detailed Solution:

The `sizeof()` operator would return size of array. To get size of string stored in array, we need to use `strlen()`. The following program prints 20 which is the size of the array.

QUESTION 10:

Point out the error, if any, in the for loop:

```
#include<stdio.h>
int main()
{
    int i = 1;
    for (;;)
    {
        printf("%d", i++);
        if (i > 10)
            break;
    }
}
```

a. ERROR: The condition part in the `for` loop is required.
b. ERROR: The two semicolons should be dropped.
c. ERROR: The `for` loop should be replaced by a `while` loop.
d. No error.

Correct Answer: d. No error.

Detailed Solution:

All components of a for loop is optional and there is no syntax error in the code.

QUESTION 11:

What is the following set of statements equivalent to?

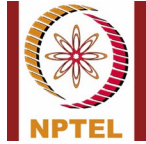```
if(x==1)
{
    x=0;
}
else if(x==0)
{
    x=1;
}
```

a. `x = 1 + x;`
b. `x = 1 - x;`
c. `x = x - 1;`
d. `x = 1 % x;`

**Correct Answer: b. `x = 1 - x;`**

**Detailed Solution:**

The above code inverts the value of `x`, i.e., performs a `NOT` operation. Using `x = 1 - x;` we can get the same effect in just a single line. If `x` is 1 then `1-x` will set the value of `x` as 0 and vice versa.

## QUESTION 12:

**What is defined by the statements given below?**

```
struct node
{
      int i;
      float j;
};
struct node *s[10];
```

a. An array, each element of which is pointer to a structure of type `node`.
b. A structure of 2 fields, each field being a pointer to an array of 10 elements.
c. A structure of 3 fields: an integer, a float, and an array of 10 elements.
d. An array, each element of which is a structure of type `node`.

Correct Answer: a. An array, each element of which is pointer to a structure of type node

Detailed Solution:

`struct node *s[10];` defines a pointer array of 10 elements of type `struct node*`.

QUESTION 13:

What needs to be called in order to exchange the values of two variables x and y in the following code?

```
void swap (int a, int b)
{
  int temp;
  temp = a;
  a = b;
  b = temp;
}
```

a. call `swap(x, y)`
b. call `swap(&x, &y)`
c. call `swap(*x, *y)`
d. `swap(x, y)` cannot be used to swap the variables

Correct Answer: d. `swap(x, y)` cannot be used as the parameters are passed by value

Detailed Solution:

`swap()` function will only swap the values of local variables `a` and `b`, however, there will be no effect on `x` and `y`, i.e. they will remain unchanged.

QUESTION 14:

What will be the output of the program if the array begins 2200 in memory?

```c
#include<stdio.h>

int main()
{
    int arr[]={12,13, 14, 1, 6};
    printf("%u, %u, %u", arr, &arr[0], &arr);
    return 0;
}
```

a. 2300, 2200, 2500
b. 2200, 2200, 2200
c. 2300, 2400, 2500
d. 2200, 2200, 2300

Correct Answer: b. 2200, 2200, 2200

Detailed Solution:

`arr`, `&arr[0]`, `&arr` all refer to base address of 2200.

QUESTION 15:

In C/ C++, if you pass an array as an argument to a function, what actually gets passed?

    a. Value of elements in array
    b. First element of the array
    c. Base address of the array
    d. Address of the last element of array

Correct Answer: c. Base address of the array

Detailed Solution:

When we pass an array as a function argument, the base address of the array will be passed.

************END***********

PROGRAMMING IN JAVA

Assignment 1

TYPE OF QUESTION: MCQ

Number of questions: 10   Total marks**: 10 × 1 = 10**

QUESTION 1:

What is the primary focus of Java programming?

    a. **Low-level optimizations**

    b. **Hardware-specific operations**

    c. **Platform independence**

    d. **Assembly language programming**

Correct Answer:

    c. **Platform independence**

Detailed Solution:

Java's primary feature is its ability to run on any platform without modification, thanks to the concept of Write Once, Run Anywhere (WORA).

QUESTION 2:

Which of the following programming principles is a key aspect of Java?

       a. Code obfuscation

       b. Platform dependence

       c. Object-oriented programming

       d. Global variables

Correct Answer:

       c. Object-oriented programming

Detailed Solution:

Java is designed based on the principles of object-oriented programming, promoting concepts like encapsulation, inheritance, and polymorphism.

QUESTION 3:

What is the last step in the Java programming process?

       a. Java Program Execution

       b. Java Program Editing

       c. Java Program Compilation

       d. C/C++ versus Java

Correct Answer:

       a. Java Program Execution

Detailed Solution:

The final step in the Java programming process is the execution of the compiled Java program on the Java Virtual Machine (JVM).

QUESTION 4:

Which of the following is NOT a Java programming tool?

> a. Eclipse
>
> b. NetBeans
>
> c. IntelliJ IDEA
>
> d. GCC

Correct Answer:

> d. GCC

Detailed Solution:

GCC (GNU Compiler Collection) is not a Java-specific tool. It is commonly used for compiling C, C++, and other languages.

QUESTION 5:

What does the term "Write Once, Run Anywhere" (WORA) imply in Java?

      a. Code reusability

      b. Platform independence

      c. Cross-compilation

      d. Dynamic typing

Correct Answer:

      b. Platform independence

Detailed Solution:

"Write Once, Run Anywhere" (WORA) is a key concept in Java, indicating that code written in Java can run on any platform without modification.

QUESTION 6:

In Java, what is used to store multiple values of the same type?

     a. Structures

     b. Pointers

     c. Arrays

     d. Lists

Correct Answer:

     c. Arrays

Detailed Solution:

Arrays in Java are used to store multiple values of the same type in a single variable.

QUESTION 7:

Which of the following is a valid identifier in Java?

    a. **123identifier**

    b. **_identifier**

    c. **#identifier**

    d. **identifier-123**

Correct Answer:

    b. **_identifier**

Detailed Solution:

Identifiers in Java can start with a letter, underscore (_), or dollar sign ($).

QUESTION 8:

What is the purpose of Java Language Subset?

     a. **To limit the capabilities of Java**

     b. **To make Java code compatible with other languages**

     c. **To define a smaller set of Java features for specific purposes**

     d. **To enhance the performance of Java programs**

Correct Answer:

     c. **To define a smaller set of Java features for specific purposes**

Detailed Solution:

Java Language Subset is a set of features tailored for specific purposes, limiting the use of certain Java features.

QUESTION 9:

What is the primary purpose of the Java Virtual Machine (JVM) in the Java programming language?

    a. Code optimization

    b. Platform independence

    c. Memory management

    d. Hardware-specific operations

Correct Answer:

    b. Platform independence

Detailed Solution:

The Java Virtual Machine (JVM) enables platform independence by interpreting Java bytecode, allowing Java programs to run on any device with a compatible JVM.

QUESTION 10:

What is emphasized during the Java Program Editing phase?

    a. **Writing platform-specific code**

    b. **Debugging the program**

    c. **Compiling the program**

    d. **Writing and modifying the source code**

Correct Answer:

    d. **Writing and modifying the source code**

Detailed Solution:

Java Program Editing involves writing and modifying the source code to implement the desired functionality before compilation.

PROGRAMMING IN JAVA

Assignment 2
TYPE OF QUESTION: MCQ

Number of questions: 10   Total marks**: $10 \times 1 = 10$**

QUESTION 1:

In Java programming an object can take many forms. This feature called _____.

- a. Abstraction
- b. Polymorphism
- c. Encapsulation
- d. Inheritance

Correct Answer:

- b. Polymorphism

Detailed Solution:

Polymorphism means "many forms", and it occurs when we have many classes that are related to each other by inheritance. Object variables (instance variables) represent the behavior of polymorphic variables in Java. It is because object variables of a class can refer to objects of its class as well as objects of its subclasses.

QUESTION 2:

Which of the following is a valid declaration of an object of class, say `NPTEL`?

    a. `NPTEL` obj = new `NPTEL` ();
    b. `NPTEL` obj = new `NPTEL`;
    c. obj = new `NPTEL` ();
    d. new `NPTEL` obj;

Correct Answer:

    a. `NPTEL` obj = new `NPTEL` ();

Detailed Solution:

The correct syntax for declaring an object is:
 <class_name> <object_name> = new <class_name>().
So the correct declaration of an object named `obj` is: `NPTEL obj = new NPTEL();`
Others are invalid declarations.

QUESTION 3:

A default constructor:

    a. **has no arguments**
    b. **has no return type**
    c. **has one argument but no return type**
    d. **has two arguments**

Correct Answer:

    a. **has no arguments**

Detailed Solution:

A default constructor is a constructor created by the compiler if we do not define any constructor(s) for a class. A constructor is called "Default Constructor" when it doesn't have any parameter. The Syntax of default constructor: <class_name>(){} . The default constructor is used to provide the default values to the object like 0, null, etc., depending on the type.
Example:

```java
class NPTEL {

  //creating a default constructor
  NPTEL() {
    System.out.println("Programming in Java");
  }

  //main method
  public static void main(String args[]) {
    //calling a default constructor
    NPTEL obj = new NPTEL();
  }
}
```

Output: "Programming in Java"

QUESTION 4:

A top-level class may have which one of the following access modifiers?

   a. package
   b. private
   c. protected
   d. public

Correct Answer:

   d. public

Detailed Solution:

At the top level only `public`, or package-private (no explicit modifier) access modifier is allowed in Java. For top level class only two access modifiers are allowed: `public` and `default`. If a class is declared as `public` it is visible everywhere. If a class is declared `default` it is visible only in same package.

QUESTION 5:

`Integer` in Java is a\an _____.

     a. Adapter class
     b. Inner class
     c. Not a class
     d. Wrapper class

Correct Answer:

     d. Wrapper class

Detailed Solution:

Byte, Short, Integer, Long, Character, Boolean, Double, Float are called wrapper class in Java.

QUESTION 6:

What is true about the `new` operator?

    a. returns a pointer to a variable
    b. creates a variable called new
    c. obtains memory for a new variable
    d. tells how much memory is available

Correct Answer:

    c. obtains memory for a new variable

Detailed Solution:

The `new` operator is used in Java to create new objects. It can also be used to create an array object. The `new` operator instantiates a class by allocating memory for a new object and returning a reference to that memory. The `new` operator also invokes the object constructor. It is used for dynamic memory allocation which puts variables on heap memory.

QUESTION 7:

Which one is not supported by OOP?

    a. **Abstraction**
    b. **Polymorphism**
    c. **Encapsulation**
    d. **Global variables**

Correct Answer:

    d. **Global variables**

Detailed Solution:

Java does not support global variables. A global variable is one declared at the start of the code and is accessible to all parts of the program. Since Java is object-oriented, everything is part of a class. The intent is to protect data from being changed. A static variable can be declared, which can be available to all instances of a class.

QUESTION 8:

Which of the following modifiers can be used to disallow a method from being overridden?

    a. **final**
    b. **transient**
    c. **volatile**
    d. **static**

Correct Answer:

    a. **final**

Detailed Solution:

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, like `PI` (3.14159...).

QUESTION 9:

Consider the following code segment

```
1  class Question{
2    public static void main(String args){
3         System.out.print("Welcome to NPTEL");
4    }
5 }
```

Identify the line number(s) where there is/are error(s) in the above code.

a. 1
b. 2
c. 3
d. 4 and 5

Correct Answer:

b. 2

Detailed Solution:

The `String` argument in the `main` method is an array hence the `args` should be changed to `args[]`.

QUESTION 10:

Which of the following is TRUE about `print()` and `println()` methods?

    a. `print()` prints in a single line only and multiple lines cannot be printed in any way.
    b. `println()`prints and then appends a line break.
    c. `println()` prints in a single line only and multiple lines cannot be printed.
    d. `print()` prints and then appends a line break.

Correct Answer:

    b. `println()`prints and then appends a line break.

Detailed Solution:

Method print() can be used to print in a single line only but multiple lines can be printed using escape sequence '\n'. Similarly, println() prints in a single line only and multiple lines can be printed using escape sequence '\n'. Method print() prints but does not appends a line break. So, option (b) println() prints and then appends a line break is the only correct option.

PROGRAMMING IN JAVA

Assignment 3

TYPE OF QUESTION: MCQ

Number of questions: 10   Total marks**: 10 × 1 = 10**

QUESTION 1:

Which of the following statement is true regarding the order of execution of constructors in an inheritance hierarchy?

    a. **Base class constructor will be called followed by the derived class constructor.**
    b. **Derived class constructor will be called followed by the base class constructor.**
    c. **Only Base class constructor will be called.**
    d. **Only derived class constructor will be called.**

Correct Answer:

    a. **Base class constructor will be called followed by the derived class constructor.**

Detailed Solution:

On object creation of derived class, first base class constructor and then the derived class constructor will be called.

QUESTION 2:

The super() method is used to:

    a. **Call constructor of friend class**
    b. **Is a declared method**
    c. **Call constructor of the parent class**
    d. **Call constructor**

Correct Answer:

    c. **Call constructor of the parent class**

Detailed Solution:

In Java programming language, the super() is a reference variable that is used to refer parent class constructors. The super can be used to call parent class's variables and methods. The super() can be used to call parent class' constructors only.

QUESTION 3:

What will be the output of the following Java program?

```java
class A {

  int i;

  void display() {
    System.out.println(i);
  }
}

class B extends A {

  int j;

  void display() {
    System.out.println(j);
  }
}

class inheritance_demo {

  public static void main(String args[]) {
    B obj = new B();
    obj.i = 1;
    obj.j = 2;
    obj.display();
  }
}
```

a. 0
b. 1
c. 2
d. Compilation Error

Correct Answer:

c. 2

Detailed Solution:

Class A & class B both contain display() method, class B inherits class A, when display() method is called by object of class B, display() method of class B is executed rather than that of Class A.

QUESTION 4:

In Java, is it possible to override a static method?

    a. Yes, we can override a static method just like we do with instance methods.
    b. No, static methods cannot be overridden because they belong to the class, not the object.
    c. It depends on whether the static method is declared as final or not.
    d. It depends on the access modifier of the static method.

Correct Answer:

    b. No, static methods cannot be overridden because they belong to the class, not the object.

Detailed Solution:

In Java, a static method is bound to the class and not to the instance. Hence, it is not part of the state of the object and doesn't participate in polymorphism and dynamic dispatch, which are necessary for method overriding.

QUESTION 5:

What is the output of the following Java program?

```java
public class Vehicle {
    public void move() {
        System.out.println("The vehicle moves");
    }
}

public class Car extends Vehicle {
    public void move() {
        System.out.println("The car moves");
    }
}

public class Main {
    public static void main(String[] args) {
        Vehicle vehicle = new Car();
        vehicle.move();
    }
}
```

a. **"The vehicle moves"**
b. **"The car moves"**
c. **The code does not compile**
d. **None of the above**

Correct Answer:

b. **"The car moves"**

Detailed Solution:

In Java, a subclass can override methods from its superclass. In this example, the Car class is overriding the move method of the Vehicle class. Since the object is instantiated as a Car, the overridden move method in the Car class is called, producing the output "The car moves"

QUESTION 6:

What is the output of the below Java program with inheritance?

```
class Sweet {

  void price() {
    System.out.print("Sweet=$10 ");
  }
}

class Sugar extends Sweet {

  void price() {
    super.price();
    System.out.print("Sugar=$20");
  }
}

public class JavaInheritance1 {

  public static void main(String[] args) {
    Sugar su = new Sugar();
    su.price();
  }
}
```

a. **Sweet=$10 Sugar=$20**
b. **Sweet=$10 Sugar=$10**
c. **Sweet=$20 Sugar=$20**
d. **Compiler error**

Correct Answer:

a. **Sweet=$10 Sugar=$20**

Detailed Solution:

Notice the use of the keyword "super". Using this keyword, you can call superclass's methods and variables.

QUESTION 7:

What is the purpose of method hiding in Java inheritance?

    a. **To prevent a subclass from inheriting methods**
    b. **To override superclass methods with new implementations**
    c. **To expose private methods of the superclass**
    d. **To define methods with the same name in both the superclass and subclass**

Correct Answer:

    d. **To define methods with the same name in both the superclass and subclass**

Detailed Solution:

Method hiding occurs when a subclass defines a static method with the same name as a static method in the superclass.

QUESTION 8:

What is the output of the following Java program?

```java
class Parent {

  String name = "parent";

  String message() {
    return "from parent";
  }
}

class Child extends Parent {

  String name = "child";

  String message() {
    return "from child";
  }
}

public class Main {

  public static void main(String[] args) {
    Parent p = new Child();
    System.out.println(p.name + " " + p.message());
  }
}
```

a. "parent from parent"
b. "child from child"
c. "parent from child"
d. "child from parent"

Correct Answer:

c. "parent from child"

Detailed Solution:

In Java, while methods are overridden (dynamic binding), variables are not overridden (static binding).
Therefore, p.name refers to the Parent class variable, and p.message() refers to the Child class method.

QUESTION 9:

Can a class be marked as both "final" and "abstract" in Java?

    a. **Yes, but only if it has no methods.**
    b. **Yes, a class can be marked as both "final" and "abstract."**
    c. **No, a class cannot be both "final" and "abstract."**
    d. **Yes, but only if it is marked as "protected."**

Correct Answer:

    c. **No, a class cannot be both "final" and "abstract."**

Detailed Solution:

A class marked as "final" cannot be extended (subclassed), while an abstract class is meant to be extended. Therefore, they are contradictory modifiers.

QUESTION 10:

In Java, is it possible to override a static method?

a. Yes, we can override a static method just like we do with instance methods.
b. No, static methods cannot be overridden because they belong to the class, not the object.
c. It depends on whether the static method is declared as final or not.
d. It depends on the access modifier of the static method.

Correct Answer:

b. No, static methods cannot be overridden because they belong to the class, not the object.

Detailed Solution:

In Java, a static method is bound to the class and not to the instance. Hence, it is not part of the state of the object and doesn't participate in polymorphism and dynamic dispatch, which are necessary for method overriding.

PROGRAMMING IN JAVA

Assignment 4

TYPE OF QUESTION: MCQ

Number of questions: 10   Total marks**: 10 × 1 = 10**

QUESTION 1:

Which is the keyword used to specify the default access modifier in java?

   a. default
   b. DEFAULT
   c. package
   d. "There is no keyword"

Correct Answer:

   d. "There is no keyword"

Detailed Solution:

If you do not specify any access modifier before a variable or method, it is considered to be declared under the `default` access modifier.

QUESTION 2:

What is the output of the Java program with access modifiers?

```java
package nptel1;

public class ProgrammingInJava {

  public String week = "FOUR";
}
// ---------------
package nptel2;

import nptel1;

public class Course {

  public static void main(String[] args) {
    ProgrammingInJava java = new ProgrammingInJava();
    System.out.println(java.week);
  }
}
```

a. **FOUR**
b. **Runtime Error**
c. **null**
d. **Compiler Error**

Correct Answer:

d. **Compiler Error**

Detailed Solution:

In order to use instance variables, one should import either a specific Class or all Classes of a Java Package. So, without using `import nptel1.*;` or `import nptel1.ProgrammingInJava;` there will be a compiler error: `ProgrammingInJava` cannot be resolved to a type.

QUESTION 3:

What is the output of the below Java Code Snippet with access modifiers?

```java
package nptel1;

public class ProgrammingInJava {

    int weeks = 12; // course duration
}

// ------------------
package nptel1;

public class Course {

    public static void main(String[] args) {
        ProgrammingInJava obj = new ProgrammingInJava();
        System.out.println("Weeks = " + obj.weeks);
    }
}
```

    a. Weeks = 0
    b. Weeks = 12
    c. No Error, blank output
    d. Compiler error

Correct Answer:

    b. Weeks = 12

Detailed Solution:

The `default` variables or methods of class can be called inside any class of the same Package.

QUESTION 4:

Which of the following is the correct representation of access modifiers in order of increasing visibility?

- a. **private < default < protected < public**
- b. **private < protected < default < public**
- c. **public < protected < default < private**
- d. **protected < default < private < public**

Correct Answer:

- a. **private < default < protected < public**

Detailed Solution:

Here's the order of the access modifiers from the least restrictive to the most restrictive:

$$public > protected > default > private$$

Thus, the correct representation of access modifiers in order of increasing visibility:

$$private < default < protected < public$$

QUESTION 5:

Which of the following package stores all the standard java classes?

    a. java.util
    b. java.lang
    c. java.java
    d. java.packages

Correct Answer:

    b. java.lang

Detailed Solution:

In Java, two packages `java.lang` package and a no-name package (also called default package) are imported by default in all the classes of Java. Default Package doesn't have a name but it is present by default and thus, it is named the default package. Java Virtual Machine imports these packages by default in Java internally. And that is the reason we are able to access all the classes of these packages. We are not required to explicitly import `java.lang` package or any of its classes like we import other packages and their classes such as `java.math` or `java.util`.

QUESTION 6:

Which of the following is/are true about packages in Java?

1. **Every class is part of some package.**
2. **All classes in a file are part of the same package.**
3. **If no package is specified, the classes in the file go into a special unnamed package.**
4. **If no package is specified, a new package is created with folder name of class and the class is put in this package.**

a. **Only 1, 2 and 3**
b. **Only 1, 2 and 4**
c. **Only 4**
d. **Only 1 and 3**

Correct Answer:

a. **Only 1, 2 and 3**

Detailed Solution:

In Java, a package can be considered as equivalent to C++ language's namespace. Thus, every class is part of some package. All classes in a file are part of the same package. If no package is specified, the classes in the file go into a special unnamed package.

QUESTION 7:

What is the output of following Java program?

```java
import static java.lang.System.*;

class ProgrammingInJava {

  public static void main(String args[]) {
    out.println("Welcome!");
  }
}
```

a. **Compiler Error**
b. **Runtime Error**
c. **Welcome!**
d. **None of the above**

Correct Answer:

c. **Welcome!**

Detailed Solution:

Static import is a feature introduced in Java programming language (versions 5 and above) that allows members (fields and methods) defined in a class as `public static` to be used in Java code without specifying the class in which the field is defined. The only time we need to pay attention to packages is when we have a name conflict . For example both, `java.util` and `java.sql` packages have a class named `Date`. So if we `import java.util.*;` and `import java.sql.*;` then the compiler will not be able to figure out which Date class do we want.

QUESTION 8:

Which of these access specifiers can be used for an interface?

    a. **Public**
    b. **Protected**
    c. **private**
    d. **All of the mentioned**

Correct Answer:

    a. **Public**

Detailed Solution:

Access specifier of an interface is either public or no specifier. When no access specifier is used then default access specifier is used due to which interface is available only to other members of the package in which it is declared, when declared public it can be used by any code.

QUESTION 9:

Which of the following is the correct way of implementing an interface NPTEL by class Java?

    a. **class Java extends NPTEL {}**
    b. **class Java implements NPTEL {}**
    c. **class Java imports NPTEL {}**
    d. **none of the mentioned**

Correct Answer:

    b. **class Java implements NPTEL {}**

Detailed Solution:

`interface` is inherited by a `class` using `implements`.

QUESTION 10:

What will be the output of the following Java program?

```java
interface calculate {
  void cal(int item);
}

class display implements calculate {
  int x;

  public void cal(int item) {
    x = item * item;
  }
}

class interfaces {
  public static void main(String args[])
        {
            display arr = new display();
            arr.x = 0;
            arr.cal(2);
            System.out.print(arr.x);
        }
}
```

   a. 0
   b. 2
   c. 4
   d. Compiler Error

Correct Answer:

   c. 4

Detailed Solution:

There is no error in the program and the `cal()` function is called successfully, computes the square of `item` and stores in `x`. Which is successfully printed.

PROGRAMMING IN JAVA

Assignment 5

TYPE OF QUESTION: MCQ

Number of questions: 10   Total marks**: 10 × 1 = 10**

QUESTION 1:

Which of the following is an incorrect statement about interfaces?

   a. **Interfaces specifies what class must do but not how it does.**
   b. **Interfaces are specified public if they are to be accessed by any code in the program.**
   c. **All variables in interface are implicitly final and static.**
   d. **All variables are static and methods are public if interface is defined public.**

Correct Answer:

   d. **All variables are static and methods are public if interface is defined public.**

Detailed Solution:

All methods and variables are implicitly public if interface is declared public.

QUESTION 2:

How do you access a static method of an interface?

   a. **Using the interface name**
   b. **Using the method name directly**
   c. **Through an object of the interface**
   d. **Through an implementation class**

Correct Answer:

   a. **Using the interface name**

Detailed Solution:

Static methods in an interface are accessed using the interface name, similar to static methods in classes.

QUESTION 3:

What is the output of the below Java program with an Interface?

```java
interface Car {
  int basePrice = 1000;
}

public class InterfaceTest2 implements Car {
  void changePrice() {
    basePrice = 2000;
    System.out.print(basePrice);
  }

  public static void main(String[] args) {
    new InterfaceTest2().changePrice();
  }
}
```

a. **1000**
b. **2000**
c. **Compiler error**
d. **None of the above**

Correct Answer:

c. **Compiler error**

Detailed Solution:

Java Interface treats its variables like constants. So, the classes implementing Interfaces, can not reassign values to the variables.

QUESTION 4:

What happens when we access the same variable defined in two interfaces implemented by the same class?

   a. **Compilation failure**
   b. **Runtime Exception**
   c. **The JVM is not able to identify the correct variable**
   d. **The interfaceName. variableName needs to be defined**

Correct Answer:

   d. **The interfaceName.variableName needs to be defined**

Detailed Solution:

Explanation: The JVM needs to distinctly know which value of variable it needs to use. To avoid confusion to the JVM interfaceName.variableName is mandatory.

QUESTION 5:

Predict the output of following Java program

```
class Test extends Exception {}

class Main {

  public static void main(String args[]) {
    try {
      throw new Test();
    } catch (Test t) {
      System.out.println("Got the Test Exception");
    } finally {
      System.out.println("Inside finally block ");
    }
  }
}
```

a. **Got the Test Exception**
   Inside finally block
b. **Got the Test Exception**
c. **Inside finally block**
d. **Compiler Error**

Correct Answer:

a. **Got the Test Exception**
   Inside finally block

Detailed Solution:

In Java, the finally is always executed after the try-catch block. This block can be used to do the common cleanup work.

QUESTION 6:

What happens if an exception is not caught in the catch block?

   a. **The finally block handles it**
   b. **The exception is ignored**
   c. **The exception is thrown to the caller method**
   d. **The program terminates immediately**

Correct Answer:

   c. **The exception is thrown to the caller method**

Detailed Solution:

If an exception is not caught in the catch block, it is propagated back to the caller method.

QUESTION 7:

What will be the output of the following Java program?

```java
class exception_handling {

  public static void main(String args[]) {
    try {
      System.out.print("Hello" + " " + 1 / 0);
    } catch (ArithmeticException e) {
      System.out.print("World");
    }
  }
}
```

a. **Hello**
b. **World**
c. **HelloWorld**
d. **Hello World**

Correct Answer:

b. **World**

Detailed Solution:

System.out.print() function first converts the whole parameters into a string and then prints, before "Hello" goes to output stream 1 / 0 error is encountered which is cached by catch block printing just "World".

QUESTION 8:

What is the output of the below Java code with Exceptions?

```java
public class ExceptionTest2 {

  public static void main(String[] args) {
    try {
      int ary[] = { 10, 20, 30 };
      int tempt = ary[4];
    } catch (ArrayIndexOutOfBoundsException e1) {
      System.out.println(e1.getMessage());
    } catch (Exception e2) {
      System.out.println("Some exception");
    }
  }
}
```

a. **Index 4 out of bounds for length 3**
b. **Index 4 out of bounds for length 3**
   Some exception
c. **Some exception**
d. **No exception occurs**

Correct Answer:

a. **Index 4 out of bounds for length 3**

Detailed Solution:

IndexOutOfBoundsException is raised by TRY block. Observe the order of catching the exceptions. Always catch a Subclass exception before a Superclass exception.

QUESTION 9:

What is the output of the Java code with FINALLY block and RETURN statement?

```java
public class ExceptionTest6 {

  static void show() {
    try {
      System.out.println("inside TRY");
      return;
    } finally {
      System.out.println("inside FINALLY");
    }
  }

  public static void main(String[] args) {
    show();
  }
}
```

a. **inside TRY**
b. **inside TRY**
   inside FINALLY
c. **inside FINALLY**
d. **Compiler error**

Correct Answer:

b. **inside TRY**
   inside FINALLY

Detailed Solution:

Even if a RETURN statement is present at the last line of TRY block, the control is not returned to the calling method. The JVM searches for the suitable FINALLY block and executes it before returning. So, the FINALLY block has higher priority than a RETURN statement.

QUESTION 10:

What is the purpose of the finally block in Java exception handling?

    a. **To handle an exception**
    b. **To catch an exception**
    c. **To clean up resources after a try block**
    d. **None of the above**

Correct Answer:

    c. **To clean up resources after a try block**

Detailed Solution:

The purpose of the finally block in Java exception handling is to clean up resources after a try block. The finally block is executed whether or not an exception is thrown in the try block. It is typically used to release resources such as open files, database connections, or network sockets that were acquired in the try block. The finally block ensures that these resources are released even if an exception is thrown, which helps prevent resource leaks and other issues. The finally block is optional, but it is good practice to use it when dealing with resources that need to be released.

PROGRAMMING IN JAVA

Assignment 6

TYPE OF QUESTION: MCQ

Number of questions: 10   Total marks**: 10 × 1 = 10**

QUESTION 1:

What is the output of the following program?

```java
public class Nptel{

  public static void main(String[] args) {
    try {
      int a = 5 / 0;
    } catch (Exception e) {
        catch (ArithmeticException a) {

      }
    }
    System.out.println("Programming In Java");
  }
}
```

a. **"Programming In Java"**

b. **Run time error**

c. **Compile time error**

d. **ArithmeticException**

Correct Answer:

c. **Compile time error**

Detailed Solution:

This first handler catches exceptions of type `Exception`; therefore, it catches any exception, including `ArithmeticException`. The second handler could never be reached. This code will not compile.

QUESTION 2:

What is the output of the following program?

```java
public class Nptel extends Thread {

  public void run() {
    for (int i = 1; i < 5; i++) {
      System.out.print(i++ + " ");
    }
  }

  public static void main(String args[]) {
    Nptel t1 = new Nptel();
    t1.run();
  }
}
```

a. **1 3**

b. **1 2 3 4**

c. **Runtime error**

d. **1 2**

Correct Answer:

a. **1 3**

Detailed Solution:

Inside the `for` loop, the increment operation `i++` is used, which is a post-increment operation. This means that the increment happens after the value of `i` is used. But since this increment operation is inside the print statement, it will increment `i` by 1 right after printing it, effectively skipping every other number. So, the output of this code will be: "1 3"

This is because the `i++` operation inside the `System.out.print` statement causes `i` to increment by an additional 1 each time through the loop, resulting in only the odd numbers between 1 and 3 being printed.

QUESTION 3:

For the program given below, what will be the output after its execution?

```java
public class Nptel extends Thread {

  public static void main(String[] args) {
    Thread thread = Thread.currentThread();
    System.out.println(thread.activeCount());
  }
}
```

a. 0

b. true

c. 1

d. false

Correct Answer:

c. 1

Detailed Solution:

`java.lang.Thread.activeCount()` returns an estimate of the number of active threads in the current thread's thread group and its subgroups which is 1 in this case since it's the only thread running the program.

QUESTION 4:

Which of the following is a correct constructor for a thread object?

a. `Thread(Runnable a, String str);`

b. `Thread(Runnable a, int priority);`

c. `Thread(Runnable a, ThreadGroup t);`

d. `Thread(int priority);`

Correct Answer:

a. `Thread(Runnable a, String str);`

Detailed Solution:

`Thread(Runnable a, String str)` creates a new `Thread` object. The others are not valid constructors to create a `Thread` object.

QUESTION 5:

What is the output of the following program?

```java
class Nptel extends Thread {

  public void run() {
    System.out.println("Running");
  }
}

public class ThreadTest {

  public static void main(String args[]) throws InterruptedException {
    Runnable r = new Nptel();
    Thread myThread = new Thread(r);
    myThread.start();
  }
}
```

    a. **Compiler Error**

    b. **"Running"**

    c. **Runtime Exception**

    d. **No output, but no error**

Correct Answer:

    b. **"Running"**

Detailed Solution:

The class `Thread` implements the `Runnable` interface, so the assignment is valid. Also, you can create a new thread object by passing a `Runnable` reference to a `Thread` constructor, is also valid. Hence, the program will compile without errors and print "Running" in the console.

QUESTION 6:

How many threads does the following program run on?

```java
public class ThreadExtended extends Thread {

  public void run() {
    System.out.println("\nThread is running now\n");
  }

  public static void main(String[] args) {
    ThreadExtended threadE = new ThreadExtended();

    threadE.start();
  }
}
```

a. 0

b. 1

c. 2

d. 3

Correct Answer:

c. 2

Detailed Solution:

There are 2 threads. Main program is also run as a thread. And, program has created one child thread. Hence, total 2 threads are there in the program.

QUESTION 7:

In the following java program, what is the NAME of the thread?

```java
class Nptel extends Thread{

    public static void main(String args[]) {
      Thread t = Thread.currentThread();
      System.out.println(t);
    }
}
```

a. thread

b. main

c. system

d. None of the above

Correct Answer:

b. main

Detailed Solution:

The name of the `Tread t` is `main` as it's the only tread that is running. It is set to `currentThread()` which is `main`.

QUESTION 8:

Which of the following line(s) of code is suitable to START a thread at #1?

```
class Nptel extends Thread {

  public static void main(String args[]) {
    /* Missing code */
    _____ // #1
  }

  public void run() {}
}
```

a. Thread t = new Thread(Nptel);

b. Thread t = new Thread(Nptel);
   t.start();

c. Nptel run = new Nptel();
   Thread t = new Thread(run);
   t.start();

d. Thread t = new Thread();
   Nptel.run();

Correct Answer:

c. Nptel run = new Nptel();
   Thread t = new Thread(run);
   t.start();

Detailed Solution:

An instance of the `Nptel` class is created. This class should implement the `Runnable` interface, which means it must have a `run` method. This `run` method contains the code that will be executed in the new thread. (`Nptel run = new Nptel();`)
 A new `Thread` object is created, passing the `Nptel` instance (`run`) to the `Thread` constructor. This tells the `Thread` that it should execute the `run` method of the `Nptel` instance in the new thread. (`Thread t = new Thread(run);`)
The `start` method of the `Thread` instance is called. This method starts the new thread and calls the `run` method of the `Nptel` instance in that new thread. (`t.start();`) So, the `run` method of the `Nptel` instance will be executed in a new thread, separate from the main thread of the application.

QUESTION 9:

What is the name of the priority of this Thread in this program?

```java
class Nptel extends Thread{

  public static void main(String args[]) {
    Thread t = Thread.currentThread();
    System.out.println(t.getPriority());
  }
}
```

a. 1

b. 4

c. 0

d. 5

Correct Answer:

d. 5

Detailed Solution:

The default priority given to a thread is 5.

QUESTION 10:

What does I/O stand for in Java?

      a. Input/Output

      b. Inheritance/Overriding

      c. Integer/Object

      d. Iteration/Observation

Correct Answer:

      a. Input/Output

Detailed Solution:

I/O stands for Input/Output in Java. It refers to the process of reading data from input sources and writing data to output destinations.

PROGRAMMING IN JAVA

Assignment 7

TYPE OF QUESTION: MCQ

Number of questions: 10   Total marks**: 10 × 1 = 10**

QUESTION 1:

Which stream does Java application uses to read data from a source, it may be a file, an array, peripheral device or socket?

      a. **InputStream**

      b. **OutputStream**

      c. **Input/OutputStream**

      d. **None of the above**

Correct Answer:

      a. **InputStream**

Detailed Solution:

`InputSteam` is used to read data from any source in Java.

QUESTION 2:

What is the primary purpose of input streams in Java?

      a. **To write data to a file.**

      b. **To read data from a file.**

      c. **To append data to a file.**

      d. **To create directories.**

Correct Answer:

      b. **To read data from a file.**

Detailed Solution:

Input streams are used to read data from a file or another source.

QUESTION 3:

Which class in Java is used to create a new directory?

   a. **FileReader**

   b. **FileWriter**

   c. **File**

   d. **Directory**

Correct Answer:

   c. **File**

Detailed Solution:

The File class is used to create directories and manipulate files.

QUESTION 4:

Which class in Java is used to read data line by line from a file?

      a. BufferedReader

      b. FileInputStream

      c. FileWriter

      d. OutputStream

Correct Answer:

      a. BufferedReader

Detailed Solution:

BufferedReader is commonly used to read data line by line from a file.

QUESTION 5:

What does the following code do?

```
FileInputStream fis = new FileInputStream("test.dat");
```

a. It creates a new file named test.dat if it does not exist and opens the file so you can write to it, if write permission is available.

b. It thows an error if the file named test.dat does not exist and opens the file, if it exists, so you can read from it and write into it, if write permission is available.

c. It creates a new file named test.dat regardless of whether it exists or not and opens the file so you can write to it, if write permission is available.

d. It creates a new file named test.dat regardless of whether it exists or not and opens the file so you can read from it and write to it, if write permission is available.

Correct Answer:

b. It thows an error if the file named test.dat does not exist and opens the file, if it exists, so you can read from it and write into it, if write permission is available.

Detailed Solution:

The provided Java code is creating a `FileInputStream` object. `FileInputStream` is a class in Java that is used for reading data from files. It's part of the Java I/O (Input/Output) package. The string "test.dat" is the path to the file. If only a filename is provided (like in this case), Java will look for the file in the current directory of the program. If the file is located in a different directory, you would need to provide the full path to the file.

Once the `FileInputStream` is created, you can use its methods to read data from the file. For example, the read method can be used to read a single byte of data from the file, and the `read(byte[] b)` method can be used to read up to `b.length` bytes of data from the file into an array of bytes.

QUESTION 6:

What is the output of this program? (Assume 'inputoutput.java' file exists in the current directory)

```java
import java.io.*;

class filesinputoutput {
    public static void main(String args[]) throws FileNotFoundException,
IOException {
        InputStream obj = new FileInputStream("inputoutput.java");
        System.out.print(obj.available());
        obj.close();
    }
}
```

a. true

b. false

c. prints number of bytes in file

d. prints number of characters in the file

Correct Answer:

c. prints number of bytes in file

Detailed Solution:

`obj.available()` returns the number of bytes.

QUESTION 7:

What will be the output of the following Java program?

```java
import java.io.*;

class Chararrayinput {

  public static void main(String[] args) {
    String obj = "abcdefgh";
    int length = obj.length();
    char c[] = new char[length];
    obj.getChars(0, length, c, 0);
    CharArrayReader input1 = new CharArrayReader(c);
    CharArrayReader input2 = new CharArrayReader(c, 1, 4);
    int i, j;
    try {
      while ((i = input1.read()) == (j = input2.read())) {
        System.out.print((char) i);
      }
    } catch (IOException e) {
      e.printStackTrace();
    }
  }
}
```

a. abc

b. abcd

c. abcde

d. none of the mentioned

Correct Answer:

d. none of the mentioned

Detailed Solution:

No output is printed. CharArrayReader object input1 contains string "abcdefgh" whereas object input2 contains string "bcde", when while((i=input1.read())==(j=input2.read())) is executed the starting character of each object is compared since they are unequal control comes out of loop and nothing is printed on the screen.

QUESTION 8:

What is the output of the following Java program?

```java
class Main {

  public static void main(String args[]) {
    final int i;
    i = 20;
    i = 30;
    System.out.println(i);
  }
}
```

a. **30**

b. **Compiler Error**

c. **Garbage value**

d. **0**

Correct Answer:

b. **Compiler Error**

Detailed Solution:

i is assigned a value twice. Final variables can be assigned values only one. Following is the compiler error "Main.java:5: error: variable i might already have been assigned".

QUESTION 9:

In order to restrict a variable of a class from inheriting to subclass, how variable should be declared?

    a. **Protected**

    b. **Private**

    c. **Public**

    d. **Static**

Correct Answer:

    b. **Private**

Detailed Solution:

By declaring variable private, the variable will not be available in inherited to subclass.

QUESTION 10:

A _____ is a type of object that organizes components in a container.

     a. Event adapter

     b. Event Handler

     c. Layout manager

     d. Grid Manager

Correct Answer:

     c. Layout manager

Detailed Solution:

A layout manager is a type of object that organizes components in a container.

PROGRAMMING IN JAVA

Assignment 8

TYPE OF QUESTION: MCQ

Number of questions: 10   Total marks**: 10 × 1 = 10**

QUESTION 1:

What does AWT stand for in Java?

      a. Applet Windowing Toolkit

      b. Abstract Window Toolkit

      c. Absolute Windowing Toolkit

      d. None of the above

Correct Answer:

      b. Abstract Window Toolkit

Detailed Solution:

The Abstract Window Toolkit (AWT) is Java's original platform-dependent windowing, graphics, and user-interface widget toolkit, preceding Swing.

QUESTION 2:

When we invoke `repaint()` for a `java.awt.Component` object, the `AWT` invokes which of the following method ?

    a. draw( )

    b. show( )

    c. update( )

    d. none of the mentioned

Correct Answer:

    c. update( )

Detailed Solution:

The `repaint()` method calls automatically `update()` method and in turn `update()` method calls `paint()` method.

QUESTION 3:

Which is/are used to create a Frame?

1. **By creating the object of `Frame` class (association)**

2. **By extending `Frame` class (inheritance)**

   a. **Only 1**

   b. **Only 2**

   c. **Both**

   d. **None**

Correct Answer:

   c. **Both**

Detailed Solution:

A `Frame` object can be created using the `Frame` class itself as well as extending the `Frame` class.

QUESTION 4:

Which AWT concept allows you to handle events such as button clicks or mouse movements?

     a. Event Handling

     b. Component Interaction

     c. Process Management

     d. GUI Processing

Correct Answer:

     a. **Event Handling**

Detailed Solution:

Event Handling in AWT enables the response to user actions, such as button clicks or mouse movements, in a graphical user interface.

QUESTION 5:

Which of the following methods can be used to change the size of a `java.awt.Component` object?

1. dimension()

2. setSize()

3. area()

4. size()

5. resetSize()

   a. 1, 2, 3 & 5

   b. 4 & 5

   c. 1, 2 & 5

   d. only 2

Correct Answer:

   d. only 2

Detailed Solution:

The method `setSize()` can only be used to change the size of a component.

QUESTION 6:

Which of the following is TRUE regarding checkbox and radio button?

> a. **Checkbox is used for single selection item whereas radio button is used for multiple selection.**
>
> b. **Checkbox is used for multiple selection items whereas radio button is used for single selection.**
>
> c. **Both are used for multiple as well as single item selection.**
>
> d. **Checkbox is the same as radio buttons.**

Correct Answer:

> b. **Checkbox is used for multiple selection items whereas radio button is used for single selection.**

Detailed Solution:

Checkbox is used for multiple selection items whereas radio button is used for single selection. For example, if a form is asking for your favorite hobbies, there might be multiple correct answers to it, in that case checkbox is preferred. And if a form is asking about gender, there must be only one true option among the multiple choices, in that case radio buttons are used.

QUESTION 7:

How is AWT different from Swing in Java?

      a. AWT is for web applications, Swing is for desktop applications

      b. AWT uses lightweight components, Swing uses heavyweight components

      c. AWT is newer than Swing

      d. AWT and Swing are synonyms, used interchangeably

Correct Answer:

      b. AWT uses lightweight components, Swing uses heavyweight components

Detailed Solution:

`AWT` components are platform-dependent and use native components, whereas `Swing` components are implemented in Java and are platform-independent.

QUESTION 8:

What is the primary purpose of AWT components in GUI programming?

     a. To perform mathematical calculations

     b. To interact with databases

     c. To create graphical user interfaces

     d. To handle network communications

Correct Answer:

     c. To create graphical user interfaces

Detailed Solution:

`AWT` components provide the building blocks for creating graphical user interfaces (GUIs) in Java applications.

QUESTION 9:

What will be the output of the Java code given below?

```java
import java.awt.*;
import java.awt.event.*;

public class ButtonExample extends Frame {
    public static void main(String[] args) {
        ButtonExample frame = new ButtonExample();
        Button b = new Button("NPTEL – Programming in Java");
        b.setBounds(30, 50, 80, 30);
        frame.add(b);
        frame.setSize(300, 200);
        frame.setLayout(null);
        frame.setVisible(true);
    }
}
```

a. A frame with a button "NPTEL – Programming in Java" at coordinates (30, 50)

b. Compilation error

c. An empty frame with no button

d. A frame with a button, but not at the specified coordinates

Correct Answer:

a. A frame with a button "NPTEL – Programming in Java" at coordinates (30, 50)

Detailed Solution:

The code creates a frame and adds a button with the label "NPTEL – Programming in Java" at coordinates (30, 50).

QUESTION 10:

What is the layout manager used in the Java code given below?

```java
import java.awt.*;
import java.awt.event.*;

public class GridLayoutExample extends Frame {
    public static void main(String[] args) {
        GridLayoutExample frame = new GridLayoutExample();
        Button b1 = new Button("Button 1");
        Button b2 = new Button("Button 2");
        Button b3 = new Button("Button 3");
        frame.add(b1);
        frame.add(b2);
        frame.add(b3);
        frame.setLayout(new GridLayout(2, 2));
        frame.setSize(300, 200);
        frame.setVisible(true);
    }
}
```

a. Border Layout

b. Flow Layout

c. Grid Layout

d. Card Layout

Correct Answer:

c. Grid Layout

Detailed Solution:

The code sets the layout manager of the 1frame to a 2x2 grid layout using `frame.setLayout(new GridLayout(2, 2))`.

PROGRAMMING IN JAVA

Assignment 9

TYPE OF QUESTION: MCQ

Number of questions: 10   Total marks**: 10 × 1 = 10**

QUESTION 1:

What is the parent class of all AWT components?

    a. **java.awt.Panel**

    b. **java.awt.Component**

    c. **java.awt.Container**

    d. **java.awt.Frame**

Correct Answer:

    b. **java.awt.Component**

Detailed Solution:

java.awt.Component is the parent class of all AWT components. It provides methods for setting and getting the properties of a component, such as its size and position.

QUESTION 2:

Which event is generated when a user clicks a button in AWT?

      a. **MouseEvent**

      b. **ActionListener**

      c. **KeyEvent**

      d. **WindowEvent**

Correct Answer:

      b. **ActionListener**

Detailed Solution:

The ActionListener event is generated when a user clicks a button in AWT. It is used to handle button clicks and perform the necessary actions.

QUESTION 3:

Which of the following architecture does the Swing framework use?

      a. MVC

      b. MVP

      c. Layered architecture

      d. Master-Slave architecture

Correct Answer:

      a. MVC

Detailed Solution:

Swing framework uses MVC architecture.

QUESTION 4:

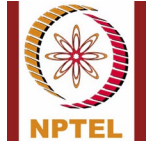A _____ is the basic class for all SWING UI components?

    a. Container

    b. JComponent

    c. Component

    d. Jbox

Correct Answer:

    b. JComponent

Detailed Solution:

A JComponent is a basic class for all SWING UI components.

QUESTION 5:

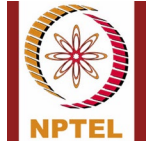**Which event is generated when a window is resized in AWT?**

   a. **WindowEvent**

   b. **ComponentEvent**

   c. **ResizeEvent**

   d. **ContainerEvent**

Correct Answer:

   b. **ComponentEvent**

Detailed Solution:

The ComponentEvent is generated when a window is resized in AWT. It is used to handle changes in the size or position of components.

QUESTION 6:

Which method is used to remove a component from a container in AWT?

      a. **remove()**

      b. **deleteComponent()**

      c. **removeComponent()**

      d. **destroy()**

Correct Answer:

      a. **remove()**

Detailed Solution:

The remove() method is used to remove a component from a container in AWT. It takes a Component object as an argument and removes it from the container.

QUESTION 7:

What is true about the following code.

```java
import javax.swing.*;

public class Nptel {
    public static void main(String[] args) {
        JFrame frame = new JFrame("My Frame");
        frame.add(new JButton("OK"));
        frame.add(new JButton("Cancel"));
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setSize(200, 200);
        frame.setVisible(true);
    }
}
```

a. **Both "OK" and "Cancel" button is added, but only "Cancel" button is visble.**

b. **Only "OK" button is added and visible, "Cancel" button is not added.**

c. **Only "Cancel" button will be added and visible, "OK" button is not added.**

d. **Code throws an ERROR.**

Correct Answer:

a. **Both "OK" and "Cancel" button is added, but only "Cancel" button is visble.**

Detailed Solution:

By default, the layout of the content pane in a JFrame is BorderLayout. Button OK is placed in the center of content pane, then button Cancel is placed in the same place. So you only can see button Cancel.

QUESTION 8:

Which of the following function is used to generate the application's top-level window?

      a. **JPanel**

      b. **JFrame**

      c. **JCombo**

      d. **JBox**

Correct Answer:

      b. **JFrame**

Detailed Solution:

JFrame is used to generate the application's top-level window.

QUESTION 9:

In Java, what is the primary purpose of a layout manager?

      a. **To manage memory allocation**

      b. **To arrange GUI components within a container**

      c. **To handle exception handling**

      d. **To control database connections**

Correct Answer:

      b. **To arrange GUI components within a container**

Detailed Solution:

A layout manager in Java is responsible for arranging and positioning GUI components within a container.

QUESTION 10:

Which layout manager divides the container into five regions: North, South, East, West, and Center?

      a. **Border Layout**

      b. **Grid Layout**

      c. **Flow Layout**

      d. **Card Layout**

Correct Answer:

      a. **Border Layout**

Detailed Solution:

Border Layout divides the container into five regions, and components can be added to each region: North, South, East, West, and Center.

PROGRAMMING IN JAVA

Assignment 10

TYPE OF QUESTION: MCQ

Number of questions: 10   Total marks**: 10 × 1 = 10**

QUESTION 1:

Which class provides methods to work with URLs?

> **a. URLConnection**
>
> **b. HttpURL**
>
> **c. NetURLg**
>
> **d. URL**

Correct Answer:

> **d. URL**

Detailed Solution:

The `URL` class provides methods to work with Uniform Resource Locators.

QUESTION 2:

What does the following code do?

```java
import javax.swing.*;

public class NPTEL extends JFrame {
    JButton button;

    public NPTEL() {
        button = new JButton("Programming in Java");
        add(button);
        setSize(300, 200);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setVisible(true);
    }

    public static void main(String[] args) {
        new NPTEL();
    }
}
```

a. **Creates a JFrame with a JButton labeled "Programming in Java"**

b. **Compiles with errors**

c. **Displays a message dialog**

d. **Creates a JPanel with a JButton labeled "Programming in Java"**

Correct Answer:

a. **Creates a JFrame with a JButton labeled "Programming in Java"**

Detailed Solution:

The code extends `JFrame` and uses the `JButton` class object to create a button with the name of "Programming in Java".
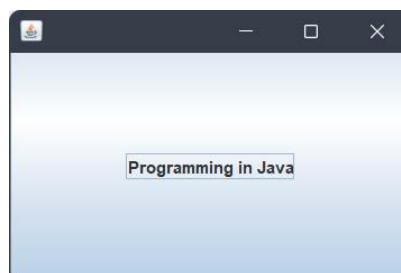
QUESTION 3:

What happens when the button in this Java code snippet is clicked?

```java
import javax.swing.*;
import java.awt.event.*;
public class NPTEL {
    public static void main(String[] args) {
        JFrame frame = new JFrame("NPTEL Java Course");
        JButton button = new JButton("Click Me");
        button.setBounds(50, 100, 100, 40);
        button.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
            JOptionPane.showMessageDialog(null, "Welcome to the course");
            }
        });
        frame.add(button);
        frame.setSize(300, 200);
        frame.setLayout(null);
        frame.setVisible(true);
    }
}
```

a. **The program exits**

b. **A message dialog with the text "Welcome to the course" is displayed**

c. **The button label changes to "Welcome to the course"**

d. **Nothing happens**

Correct Answer:

b. **A message dialog with the text "Welcome to the course" is displayed**

Detailed Solution:

The code creates a button with label "Click Me" and in the frame titled "NPTEL Java Course". A action listener is defined that opens a new message dialog with the text "Welcome to the course" when the button is clicked.
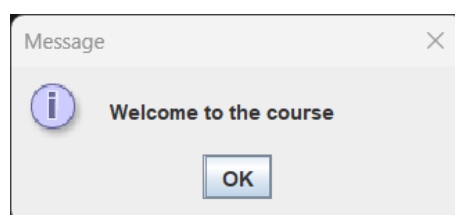
QUESTION 4:

What GUI component is used to input the password in this Java code snippet?

```java
import javax.swing.*;
public class NPTELLoginGUI extends JFrame {
    JTextField username;
    JPasswordField password;
    JButton loginButton;
    public LoginGUI() {
        username = new JTextField();
        password = new JPasswordField();
        loginButton = new JButton("Student Login");
        add(username);
        add(password);
        add(loginButton);
        setSize(300, 200);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLayout(null);
        setVisible(true);
    }
    public static void main(String[] args) {
        new LoginGUI();
    }
}
```

a. JTextField

b. JButton

c. JtextArea

d. JPasswordField

Correct Answer:

d. JPasswordField

Detailed Solution:

The GUI component which is used to input the password field is `JPasswordField`.

QUESTION 5:

What does this Java code snippet create?

```java
import javax.swing.*;
public class NPTEL {
    public static void main(String[] args) {
        JFrame frame = new JFrame("NPTEL Java Course");
        String[] colors = {"Red", "Green", "Blue"};
        JComboBox<String> comboBox = new JComboBox<>(colors);
        comboBox.setBounds(50, 50, 90, 20);
        frame.add(comboBox);
        frame.setSize(300, 200);
        frame.setLayout(null);
        frame.setVisible(true);
    }
}
```

a. A frame with a list of colors displayed as buttons

b. A frame with a combo box containing options "Red", "Green", and "Blue"

c. A frame with radio buttons for selecting colors

d. A frame with checkboxes for selecting colors

Correct Answer:

b. A frame with a combo box containing options "Red", "Green", and "Blue"

Detailed Solution:

The code snippet creates a `JComboBox` with the options "Red", "Green", and "Blue", and adds it to a `JFrame`.
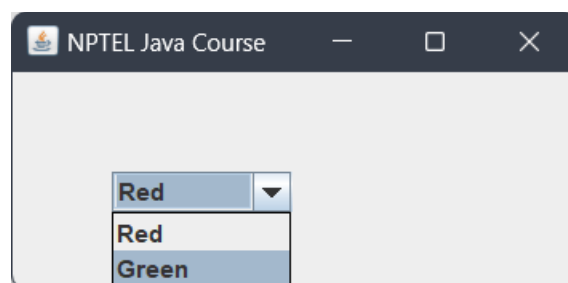
QUESTION 6:

What does this Java code snippet do?

```java
import java.net.*;

public class NPTEL {
    public static void main(String[] args) {
        try {
            InetAddress address = InetAddress.getByName("nptel.ac.in");
            System.out.println("Host Name: " + address.getHostName());
            System.out.println("IP Address: "+ address.getHostAddress());
        } catch (Exception e) {
            System.out.println(e);
        }
    }
}
```

a. Just prints the IP address of the local machine

b. Prints the IP address and host name of the local machine

c. Prints the IP address and host name of "nptel.ac.in"

d. Just prints the IP address of "nptel.ac.in"

Correct Answer:

c. Prints the IP address and host name of "nptel.ac.in"

Detailed Solution:

The code snippet retrieves and prints the IP address and host name of the specified domain "nptel.ac.in".
Output:

```
Host Name: nptel.ac.in
IP Address: 216.239.34.21
```

QUESTION 7:

What does this Java code snippet do?

```java
import java.sql.*;

public class NPTEL {
    public static void main(String[] args) {
        try {
            Class.forName("com.mysql.jdbc.Driver");
            Connection con = DriverManager.getConnection(
                "jdbc:mysql://localhost:3306/mydatabase",
                "username", "password");
            Statement stmt = con.createStatement();
            ResultSet rs = stmt.executeQuery("SELECT * FROM employees");
            while (rs.next()) {
                System.out.println(rs.getString(1)+" "+rs.getString(2));
            }
            con.close();
        } catch (Exception e) {
            System.out.println(e);
        }
    }
}
```

a. **Connects to a MySQL database, retrieves data from the "employees" table, and prints it**

b. **Inserts data into the "employees" table of a MySQL database**

c. **Deletes data from the "employees" table of a MySQL database**

d. **Updates data in the "employees" table of a MySQL database**

Correct Answer:

a. **Connects to a MySQL database, retrieves data from the "employees" table, and prints it**

Detailed Solution:

The code snippet establishes a connection to a MySQL database, executes a SELECT query to retrieve data from the "employees" table, and prints the results.

QUESTION 8:

What does this Java code snippet do?

```java
import java.sql.*;
public class NPTEL {
    public static void main(String[] args) {
        try {
            Class.forName("com.mysql.jdbc.Driver");
            Connection con = DriverManager.getConnection(
                "jdbc:mysql://localhost:3306/mydatabase",
                "username", "password");
            String query="INSERT INTO employees (name,age)VALUES(?, ?)";
            PreparedStatement pstmt = con.prepareStatement(query);
            pstmt.setString(1, "John");
            pstmt.setInt(2, 30);
            int rowsAffected = pstmt.executeUpdate();
            System.out.println(rowsAffected + " row(s) inserted.");
            con.close();
        } catch (Exception e) {
            System.out.println(e);
        }
    }
}
```

a. Retrieves data from the "employees" table

b. Inserts a new employee record into the "employees" table

c. Updates employee records in the "employees" table

d. Deletes employee records from the "employees" table

Correct Answer:

b. Inserts a new employee record into the "employees" table

Detailed Solution:

The code snippet inserts a new employee record with name "John" and age 30 into the "employees" table of a MySQL database using a PreparedStatement.

QUESTION 9:

What additional code is needed at #1 to complete this Java program to send a message?

```java
import java.net.*;
public class NPTEL {
    public static void main(String[] args) {
        try {
            DatagramSocket socket = new DatagramSocket();
            InetAddress address = InetAddress.getByName(
                "localhost");
            String message = "Hello, UDP Server!";
            DatagramPacket packet = new DatagramPacket(
                message.getBytes(),
                message.length(),
                address,
                9876);
            // #1 (MISSING CODE)
            socket.close();
        } catch (Exception e) {
            System.out.println(e);
        }
    }
}
```

    a. socket.recieve(packet);

    b. Socket.Send(packet);

    c. Socket.send(packet);

    d. socket.send(packet);

Correct Answer:

    d. socket.send(packet);

Detailed Solution:

Java is case sensitive. A `socket.send()` function needs to be called to send the packet to the server.

QUESTION 10:

What is missing in this Java program?

```java
public class NPTEL {
    public static void main(String[] args) {
        try {
            Class.forName("com.mysql.jdbc.Driver");
            Connection con = DriverManager.getConnection(
                "jdbc:mysql://localhost:3306/mydatabase",
                "username", "password");
            String query="INSERT INTO employees(name, age)VALUES (?, ?)";
            PreparedStatement pstmt = con.prepareStatement(query);
            pstmt.setString(1, "John");
            pstmt.setInt(2, 30);
            int rowsAffected = pstmt.executeUpdate();
            System.out.println(rowsAffected + " row(s) inserted.");
            con.close();
        } catch (Exception e) {
            System.out.println(e);
        }
    }
}
```

a. Retrieving the number of employees

b. Deleting employee records

c. Updating employee records

d. Importing necessary JDBC libraries

Correct Answer:

d. Importing necessary JDBC libraries

Detailed Solution:

The program lacks the import statements for the necessary JDBC libraries like `java.sql.*`.

PROGRAMMING IN JAVA

Assignment 11
TYPE OF QUESTION: MCQ

Number of questions: 10   Total marks**: 10 × 1 = 10**

QUESTION 1:

What is the correct order to close database resources?

   a. **Connection then Statement then ResultSet**

   b. **ResultSet then Statement then Connection**

   c. **Statement then Connection then ResultSet**

   d. **Statement then ResultSet then Connection**

Correct Answer:

   b. **ResultSet then Statement then Connection**

Detailed Solution:

When manually closing database resources, they should be closed in the reverse order from which they were opened. This means that the ResultSet object is closed before the Statement object and the Statement object is closed before the Connection object.

QUESTION 2:

How many types of JDBC drivers there?

     a. 3

     b. 4

     c. 8

     d. 10

Correct Answer:

     b. 4

Detailed Solution:

There are 4 types of JDBC drivers: Type-1 driver or JDBC-ODBC bridge driver, Type-2 driver or Native-API driver, Type-3 driver or Network Protocol driver, Type-4 driver or Thin driver.

QUESTION 3:

What is the correct sequence to create a database connection?

  i.   **Import JDBC packages.**
  ii.   **Open a connection to the database.**
  iii. **Load and register the JDBC driver.**
  iv.   **Execute the statement object and return a query resultset.**
  v.   **Create a statement object to perform a query.**
  vi.   **Close the resultset and statement objects.**
  vii. **Process the resultset.**
  viii. **Close the connection.**

   a. **i, ii, iii, v, iv, vii, viii, vi**

   b. **i, iii, ii, v, iv, vii, vi, viii**

   c. **ii, i, iii, iv, viii, vii, v, vi**

   d. **i, iii, ii, iv, v, vi, vii, viii**

Correct Answer:

   b. **i, iii, ii, v, iv, vii, vi, viii**

Detailed Solution:

To create a database connection in Java, we must follow the sequence given below:
   1. Import JDBC packages.
   2. Load and register the JDBC driver.
   3. Open a connection to the database.
   4. Create a statement object to perform a query.
   5. Execute the statement object and return a query resultset.
   6. Process the resultset.
   7. Close the resultset and statement objects.
   8. Close the connection.

QUESTION 4:

Which of the following is correct about connection pooling?

   a. **Application server like WebLogic, WebSphere, jBoss and Tomcat provides the facilities to configure connection pooling.**

   b. **components like Apache Commons DBCP Component can be used to configure connection pooling.**

   c. **Both of the above.**

   d. **None of the above.**

Correct Answer:

   c. **Both of the above**

Detailed Solution:

If you use an application server like WebLogic, WebSphere, jBoss, Tomcat. , then your application server provides the facilities to configure for connection pooling. If you are not using an application server then components like Apache Commons DBCP Component can be used.

QUESTION 5:

Which of the following is used to test the operation?

      a. **JDBC API**

      b. **JDBC Driver manager**

      c. **JDBC Test suite**

      d. **JDBC-ODBC Bridge Drivers**

Correct Answer:

      c. **JDBC Test suite**

Detailed Solution:

JDBC Test suite: It is used to test the operation(such as insertion, deletion, updation) being performed by JDBC Drivers.

QUESTION 6:

The JDBC architecture consists of _____ to access a database.

       a. three-tier processing models

       b. two-tier processing models

       c. two-tier and three-tier processing models

       d. None of the above

Correct Answer:

       c. two-tier and three-tier processing models

Detailed Solution:

The JDBC architecture consists of two-tier and three-tier processing models to access a database.

QUESTION 7:

Which of these obtains a Connection?

> a. **Connection.getConnection(url)**
>
> b. **Driver.getConnection(url)**
>
> c. **DriverManager.getConnection(url)**
>
> d. **new Connection(url)**

Correct Answer:

> c. **DriverManager.getConnection(url)**

Detailed Solution:

Option C is the correct answer because DriverManager is the class used in JDBC to get a Connection.

QUESTION 8:

Which class provides methods to create a client-side socket in Java?

      a. ServerSocket

      b. NetSocket

      c. Socket

      d. ClientSocket

Correct Answer:

      c. Socket

Detailed Solution:

The `Socket` class in the `java.net` package is used to create client-side sockets.

QUESTION 9:

What does JDBC stand for?

> **a. Java DataBase Connectivity**
>
> **b. Java DataBase Connection**
>
> **c. Java DataBase Control**
>
> **d. Java DataBase Connector**

Correct Answer:

> **a. Java Database Connectivity**

Detailed Solution:

JDBC stands for Java DataBase Connectivity.

QUESTION 10:

Which method can be used to query for a single object using JdbcTemplate?

      a. **queryForObject()**

      b. **queryForList()**

      c. **query()**

      d. **singleQuery()**

Correct Answer:

      a. **queryForObject()**

Detailed Solution:

The queryForObject() method of JdbcTemplate is used to query for a single object.