

UNIT

1

OBJECT ORIENTED THINKING AND INHERITANCE

Marketed by:



PART-A

SHORT QUESTIONS WITH SOLUTIONS

Q1. Define method overriding.

Answer :

Model Paper-I, Q1(a)

Method overriding is a phenomenon in which a method in subclass is similar to the method in superclass. The return type and signature of a subclass method matches with the return type and signature of superclass method. The advantage of this method is that, it provides an implementation which is already provided by its superclass. This method can be used for runtime polymorphism. When the method of subclass is called the superclass's method is not referred and it is hidden.

Q2. What is an exception?

Answer :

Model Paper-II, Q1(a)

If the normal flow of the program is disrupted, during the program execution, an event is occurred which is called as "Exception".

Exceptions are generally used to manage errors.

It has an object called exception object that holds the error information. This information includes, the type and state of the program when the error occurred.

To throw an exception, an exception object is created and is then handed over to the runtime system.

Q3. List the features of Java.

Answer :

The following are the various features of Java,

1. Object-oriented
2. Compiled and interpreted
3. Platform-independent and portable
4. Distributed
5. Robust and secure
6. Familiar, simple and small
7. Multithreaded and interactive
8. Dynamic and extensible
9. High performance.

Q4. Define variable.

Answer :

Model Paper-III, Q1(a)

A variable is the name given to a unit/memory location that stores data value of the variable. The name given to the variable is known as Identifier.

The variable value may change several times during the execution of program. Each variable is associated with its scope that depicts the life time and visibility of the variables.

1.2

Q5. Define type casting.**Answer :**

'Type casting' is an explicit conversion of a value of one type into another type. And simply, the data type is stated using parenthesis before the value. Type casting in Java must follow the given rules,

1. Type casting cannot be performed on Boolean variables. (i.e., boolean variables can be cast into other data type).
2. Type casting of integer datatype into any other datatype is possible. But, if the casting into smaller type is performed, it results in loss of data.
3. Type casting of floating point types into other float types or integer type is possible, but with loss of data.
4. Type casting of char type into integer types is possible. But, this also results in loss of data, since char holds 16-bits the casting of it into byte results in loss of data or mixup characters.

Q6. List the advantages of arrays.**Answer :**

Advantages of arrays are as follows,

1. Array is the simplest kind of data structure.
2. It is relatively easy to create, understand and implement arrays.
3. In arrays direct access to any element is possible. However, modifications done to one element does not effect the other.
4. Arrays have the capability of linking data together, especially with multiple dimension.
5. Every array element can be accessed at constant time.
6. Array variables are capable of storing more number of values.

Q7. List the types of operators.

Model Paper-I, Q1(b)

Answer :

The different types of operators that are present in java are as follows,

1. Arithmetic operators
2. Relational operators and logical operators
3. Assignment operators
4. Bitwise operators
5. Shift operators
6. Ternary operator (?).

Q8. Define an expression.

Model Paper-II, Q1(f)

Answer :

Expression: An expression can be defined as a combination of operators, variables and constants. These are referred as constituents of an expression.

Example

```
c = a + b;
```

```
a = 5;
```

```
b = a * c;
```

Q9. Define class.**Answer :**

A class can be defined as a template that groups data and its associated functions. The class contains two parts namely,

- (a) Declaration of data variables
- (b) Declaration of member functions.

UNIT-1 Object Oriented Thinking and Inheritance

The data members of a class explains about the state of the class and the member function explains about the behavior of the class. There are three types of variables available for a class. They are,

- (i) Local variables
- (ii) Instance variables
- (iii) Class variables.

Q10. Write in brief about string.

Answer :

A sequence of characters together is called as a string. Strings in java are class objects. They are implemented through the use of classes such as String and StringBuffer. They are reliable and predictable when compared to that of other languages. A string can be declared as follows,

```
String stringName;  
StringName = new String("string");  
or  
String stringName = new String("string");
```

Q11. Define final class.

Answer :

Model Paper-III, Q1(b)

Final Class: The class which cannot be inherited or extended is called final class. A final class can be created by proceeding the class name with final keyword in its definition. This is done to prevent it from being inherited. If these classes are tried to be inherited then the compiler generates errors. Declaring a class as final will automatically declare its methods as final.

Example

```
final class Baseclass  
{  
    System.out.println("Baseclass");  
}  
class Derivedclass extends Baseclass //error  
{  
    System.out.println("Derivedclass");  
}
```

Q12. What is polymorphism?

Answer :

Polymorphism: Polymorphism is an important concept in OOP. It is derived from the Greek word poly (multiple) and morphism (forms) which together mean multiple forms. It is a method through which an operation/function can take several forms based on the type of objects. An individual operator/function can be used in multiple ways.

Types of Polymorphism: There are two types of polymorphism, they are,

- (i) Adhoc polymorphism
- (ii) Pure polymorphism.

Q13. Write a short note on Java class libraries.

Answer :

The mostly used built-in methods are println() and print() methods. They are contained in System.out.println which is a class of java by default.

Java language depends upon various built in class libraries that will in turn contain builtin methods which support I/O, string handling, networking and graphics. These classes also support GUI. Therefore, Java is considered as a combination of itself and standard classes.

These classes provide the functionality of Java.

UNIT

2

PACKAGES, INTERFACES AND STREAM BASED I/O (java.io)



PART-A

SHORT QUESTIONS WITH SOLUTIONS

Q1. Define package.

Model Paper-I, Q1(c)

Answer :

Package is a mechanism that organizes classes and interfaces. It groups them based on their functionality. It acts as a container for the classes. Packages are of two types namely Java API Package and user-defined packages. Java API consists of various packages which contain classes. Packages involve in access control mechanism of Java. A class which is define in a package will be private to the package only. It cannot be accessed by a code which is beyond the package.

Q2. Define CLASSPATH.

Model Paper-II, Q1(b)

Answer :

'CLASS PATH' can be defined as a statement which gives path or location of the class which is being currently used. When a package is created by a programmer and if the programmer is trying to execute one of the classes in it, then it is important to know where the Java run-time system looks for the package.

Q3. What is an interface?

Model Paper-III, Q1(c)

Answer :

Interfaces are similar to that of classes. It is specifically used to implement the concept of multiple inheritance. It does not contain instance variables. Interface methods are abstract and contains information about name, parameters and return type. They do not have implementation details and are publicly by default. An interface is also allowed to contain methods and variables which are only abstract and final respectively. The class which use the interface are responsible for providing implementation code for these methods.

Q4. Define stream.

Answer :

Stream is a flow of data between source (such as file on harddisk, keyboard etc.,) and destination (such as computer screen, file or socket) through a program. If the data flows from the source to a program it is called input stream. If the data flows from a program to destination, it is called output stream. They can be handled using the classes and interfaces provided by java.io package.

Q5. List the class of reader.

Answer :

Model Paper-I, Q1(d)

Class	Description
1. BufferedReader	It is used for reading the text from the character input stream and then buffering them for efficient reading.
2. CharArrayReader	It is a buffered character input stream used to keep track of the line numbers.
3. InputStreamReader	It is used for implementing the character buffer which is used as character input stream.
4. FilterReader	It is used to read bytes and decode them to characters by using a charset.
5. FileReader	It is used to read the character files.
6. PushbackReader	It is used to read the filtered character streams.
5. PipedReader	It is a character stream reader that is used to allow the characters to be pushed back to the stream.
6. StringReader	It is a piped character input stream.
	It is a character stream extracted from a string.



Q6. What is serializable interface?

Answer :

Objects of a class that implements Serializable interface can only be stored and retrieved i.e., a class that implements this interface is a serializable class. This interface does not define any methods. It is used to indicate that the objects of this class are serializable. Subclasses of a class that implements serializable interface are also serializable. Serialization does not store the variables that are declared as transient and static.

Q7. Define enumerated datatype.

Answer :

Enumerated data type can be defined as a data type which allows the user to create own data type and also define values to the variables of this type. It is a user-defined data type. An enumerated data type can be created by using a keyword 'enum'. The syntax of enumerated data type is as follows,

Syntax

```
enum datatype_name
{
    value1, value2, ...., valuen
};
```

Q8. What is Autoboxing?

Model Paper-II, Q1(g)

Answer :

The automatic process of encapsulating or boxing a primitive type into its equivalent type wrapper (i.e., Double, Float, Long, Integer, Short, Byte, Character or Boolean), when its object is required is called autoboxing. In other words, autoboxing is an automatic conversion of primitive type into an object. It avoids wrapping up of primitive data types through manual or explicit construction objects. Instead, it allows to assign the value of a primitive type to the reference of a type wrapper.

Q9. Discuss in brief about generics.

Model Paper-III, Q1(d)

Answer :

Generics is a concept in Java that enables a programmer to define and apply an algorithm irrespective of the data type. That same algorithm can be applied multiple times on many data types without enforcing additional effort. Using generic it is possible to create classes, interfaces and method that can be executed in a type-safer manner independent of any specific type of data. Such methods/classes are referred as generic methods/classes. The major advantage of using generic is that it executes the program in type-safe mode. Moreover, generic expands the programmer ability in reusing the code.

Syntax: class className<Datatype.param 1, Datatype param 2,.....Datatype param n>

```
{  
:  
}
```

Now, the way of declaring reference to generic class is,

```
classname<datatype arg 1, datatype arg 2.....datatype arg n> variable name = new classname<datatype arg 1, datatype  
arg 2.....datatype arg n>(cons arg 1, cons arg n);
```

Q10. How a generic method is created?

Answer :

Method that is declared inside a generic class can be use the type parameters of the class, thereby making the method a "generic method". This method can be declared with at least one type parameter of the respective method itself. In addition to this, it is also possible to create a generic method which is defined within a non-generic class. The primary advantage of generic method is that, the method ensures type safety.

Syntax for Declaring Generic Method: <list_of_type parameters> return_type name_of_method (list_of_parameters)

The list of type parameters are separated by comma and are placed before the return type.

PART-A**SHORT QUESTIONS WITH SOLUTIONS**

Q1. List the benefits of exception handling.

Model Paper-I, Q1(e)

Answer :

The benefits of exception handling are as follows,

- (i) Exception handling can control run time errors that occurs in the program.
- (ii) Exception handling can avoid abnormal termination of the program and also shows the behavior of program to users.
- (iii) Exception handling can provide a facility to handle exceptions, throws message regarding exception and completes the execution of program by catching the exception.
- (iv) Exception handling can separate the error handling code and normal code by using try-catch block.
- (v) Exception handling can produce the normal execution flow for a program.

Q2. What is termination model?

Answer :

In this model of exception handling, the programmer will have to explicitly invoke the same method in which the error was occurred and was transferred to the catch block so that the error can be handled. This explicit invocation is required, because few programming languages that use this kind of model do not allow the control to return back to the point where the error was occurred.

Q3. Discuss in brief about try statement.

Model Paper-I, Q1(f)

Answer :

The try statement can handle the runtime exceptions generated during the execution of a Java program. This can be done by enclosing the code which can generate exceptions in the try block. It is necessary for every try block to have one or more catch blocks. This catch block displays the type of exception which a programmer intends to catch.

Syntax

```
try
{
    //block of code which raises an exception
}
catch block
```

Q4. Write about throw statement.

Answer :

Model Paper-II, Q1(c)

'Throw' is a Java keyword used in exception handling. Generally a try block checks if any exceptions are raised. And when an error occurs, it throws the error which is caught by the catch statement. Basically, the exceptions thrown by the Java run-time system are being caught, but throw statement allows a program to throw an exception explicitly.

Syntax

```
throw ThrowbleInstance;
```

The ThrowbleInstance should be an object of either Throwble class or any of its subclass. The int, char, String etc., type of objects cannot be thrown.

Q5. Differentiate process based multitasking and thread based multitasking.

Answer :

Process-based Multitasking	Thread-based Multitasking
1. More overhead is required in multitasking processes.	1. Less overhead is required in multitasking threads.
2. Each process has its own address space. Therefore, they are heavy weight.	2. Threads share the same address space and the same process. Therefore, they are light weight.
3. Interprocess communication is expensive and limited.	3. Interthread communication is expensive.
4. Context switching between processes is costly.	4. Context switching between threads is cheaper.
5. It is not controlled by Java.	5. It is controlled by Java.

Q6. Define thread.

Answer :

Model Paper-II, Q1(f)

Thread can be defined as a set of executable instructions that are executed independently. A program can be divided into multiple subprograms and each subprogram is called a thread. Every individual thread is executed separately, thereby decreasing the execution time of a program.

Q7. How a runnable interface is implemented?

Answer :

Implementing Runnable Interface: The simple and easiest way for creating a thread is to create a class which implements the 'Runnable' interface. It is possible to construct a thread irrespective of object which implements 'Runnable'. This interface hides a unit of executable code. The implementation of 'Runnable' interface is done by implementing a single method called run(), which contains the creation code similar to main thread, run() can invoke other methods, use other classes and declare variables.

Syntax: public void run()

Q8. Define synchronization.

Answer :

Model Paper-III, Q1(e)

Synchronization can be defined as a process of enabling single thread to access shared resources. In multithreaded programming, synchronization threads are essential. When threads are attempting to access the shared resources synchronization, data can be altered and result in incorrect output. To overcome this situation, synchronization threads can be utilized.

Q9. Write about notify() and notifyall().

Answer :

notify(): This method is used to resume the initial thread which is in sleep mode.

Syntax: final void notify()

notifyall(): This method is used to resume all the threads that are in sleep mode. These threads can be executed depending on its priority.

Syntax: final void notifyall().

Q10. Write brief about wait().

Answer :

wait(): This method is used to send the invoking thread into sleep mode. The thread which is sent to sleep mode can be resumed with the help of notify() or notifyall() method. In addition to this, user can set time to the thread by declaring the timer as an argument for wait() method. After the completion of timer, the thread can be resumed automatically.

Model Paper-III, Q1(f)

Syntax: final void wait().

UNIT

4

THE COLLECTIONS FRAMEWORK (java.util)



PART-A

SHORT QUESTIONS WITH SOLUTIONS

Q1. Define collection interfaces.

Answer :

Model Paper-I, Q1(g)

The collection framework of Java is built on a standard set of interfaces called collection interfaces. They are basically defined for implementing collection mechanism. The various interface of collection framework are as follows,

1. Collection interface
2. List interface
3. Queue interface
4. Dequeue interface
5. Set interface
6. Sortedset interface.

Q2. List various collection classes.

Answer :

Model Paper-II, Q1(d)

The collection classes defined by collections framework are,

1. AbstractCollection
2. AbstractList
3. AbstractSequentialList
4. LinkedList
5. ArrayList
6. AbstractSet
7. HashSet
8. LinkedHashSet
9. TreeSet
10. Priority Queue
11. Array Dequeue.

Q3. Write a short notes on HashSet class.

Answer :

The HashSet inherits AbstractSet class and Set interface. The HashSet class is used to store the collection created by it into the hash table that stores the unique key. These keys are converted into hash code by applying a hashing technique. This hash code is then used to retrieve the data associated with the key. The hash table does not store the elements in any sorted order. Hashing is useful to perform the operations such as add(), remove(), size() and contains().

Q4. Define Iterator.

Answer :

Model Paper-III, Q1(g)

In computer vocabulary, iterator means an object with which we can travel all through a range of elements. In Java, Iterator is an interface that allows programmers to access a collection.

Following is the signature of iterator interface,

```
public Interface Iterator
```

Q5. List out the map interfaces.**Answer :**

There are four map interfaces,

1. Map
2. Sorted map
3. Navigable map
4. Map.Entry

Q6. What are the constructors used in TreeMap?

Model Paper-I, Q1(h)

Answer :

The constructors which are in TreeMap are as follows,

1. **TreeMap():** It creates an empty TreeMap sorted as per the key's natural order(ascending order).
2. **TreeMap(Map m1):** It creates a TreeMap object with the elements of another Map class as per key's natural order.
3. **TreeMap(Comparator comp):** It creates an empty TreeMap object with the sorting order of the Comparator specified by "comp".
4. **TreeMap(SortedMap m1):** It creates a TreeMap object with the same elements of the SortedMap specified by "m1" with the same sorting order of "m1".

This class does not define its own methods.

Q7. Write about comparator interface.

Model Paper-II, Q1(i)

Answer :

In general, a comparator compares two items of data. Java also means the same. In Java, Comparator is used to compare two stored objects in a Set or a Map etc.

The java.util.Comparator interface is used to dictate in which order we want the sorting of the objects of a data structure. Generally, Comparator is not needed if there's a natural sorting order. Natural sorting order means alphabetical order-A before B or 1 before 2 etc. A TreeSet or a TreeMap, by default, gives a natural sorting order, but if we want a different order, then we require comparator.

Q8. What is Arrays class?

Model Paper-III, Q1(h)

Answer :

Arrays is a class in java.util package introduced from JDK 1.5 version, which provides many static methods to do operations on arrays like filling, comparing, sorting and searching. Earlier to the introduction of Arrays, these operations were done from scratch. Now, the coding has become easier with Arrays. Many static methods are overloaded.

Following is the class signature,

```
public class Arrays extends Object
```

Q9. Write any two methods of dictionary.**Answer :**

The two methods of dictionary are,

1. **elements():** The element() method returns an enumeration of values in the dictionary. Its syntax is,

Enumeration< v > elements()

 Where, v is the type of the values.
2. **get():** The get() method returns the object that has the value of the given key. It returns null object if the key is not found in the dictionary. Its syntax is,

V get(Objectkey)

Q10. Write a short notes on StringTokenizer Class.**Answer :**

StringTokenizer Class separator string/text into tokens delimited by \s,\t,\n,\r and \f. It takes a string as input and parses it into token. This method is known as parsing. StringTokenizer provides lexer/scanner since it is the initial step in the process of parsing.

UNIT

5

GUI PROGRAMMING WITH SWING, EVENT HANDLING AND APPLETS



PART-A

SHORT QUESTIONS WITH SOLUTIONS

Q1. Define swing.

Answer :

Model Paper-I, Q1(i)

Swing is a set of classes that provides more powerful and flexible functionalities when compared to AWT components. Besides components such as buttons, check boxes and labels. Swing adds other components such as tabbedpanes, scrollpanes, trees and tables. And each component in Swing have more capabilities. For example, a button can have an image and a text string with it.

Q2. Write the differences between swing and AWT.

Answer :

Swing	AWT
1. Swing components are lightweight components, i.e., they are platform-independent.	1. AWT components are heavy-weight components. This means that AWT components are platform-specific.
2. The look and feel of each component is defined by swing, not by the platform. Hence, the working of swing components is consistent across any operating system or platform.	2. The look and feel of each component is determined by the platform.
3. Swing components are transparent.	3. AWT components are always rectangular and opaque.

Q3. Discuss in brief about light weight container.

Model Paper-II, Q1(e)

Answer :

Light-weight Containers: JPanel is one of the light-weight containers that is defined in swing. The light-weight containers inherit the 'JComponent' class and are generally used in organizing and managing groups of components that are related to each other. This is possible because a container can have a light-weight container within it.

Q4. Write about checkbox.

Model Paper-III, Q1(i)

Answer :

A JCheckBox is a swing GUI component that provides the functionality of a checkbox. Checkboxes have two states i.e., selected or not (true or false).

Checkboxes are created using JCheckBox class which is subclass of JToggleButton. Hierarchy of JCheckBox and JRadioButton is shown in the figure below. Here is an example of creating checkboxes and formatting the text as per the checkbox selected.

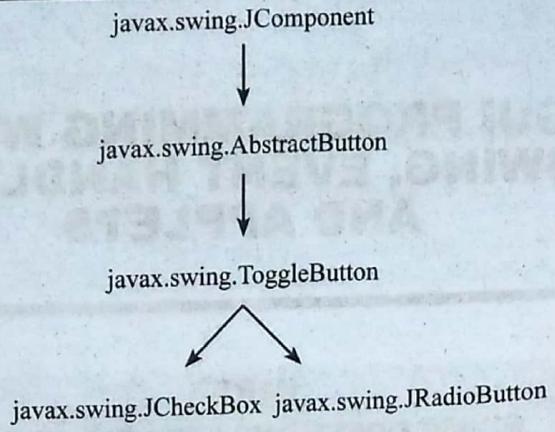


Figure: Hierarchy of JCheckBox and JRadioButton

Model Paper-I, Q1(i)

Q5. Define event.**Answer :**

An event is an action performed by the user. In the delegation event model, an event is an object that shows a status i.e., state changes in a source. Events are generated when the user interacts with the elements in a graphical user interface. Some of the actions that generate events are pressing a button, entering the data via keyboard, selecting an item in a list, clicking the mouse etc. These are all the events that are generated by a user by interacting with a user interface.

Q6. Write about adapter class.**Answer :**

It is time consuming to override all the methods of an interface to handle particular event. For example, to close a window the user needs to override all the abstract methods of the Window Listener interface. Even though we need to override only one method namely windowClosing() we are forced to override all the remaining methods of this interface.

Model Paper-II, Q1(j)

Q7. Discuss in brief about applets.**Answer :**

An applet is a Java program that is embedded in HTML document and run with the help of Java enabled browsers such as, Internet Explorer. In other words, an applet is a Java program that runs in a browser. Unlike Java applications, applets don't have a main() method. All applets inherit the superclass 'Applet'. An Applet class contains several methods that helps to control the execution of an applet. All applets must import java.applet and java.awt packages.

Q8. Differentiate between application and applet.**Answer :**

Application	Applet
1. Applications are stand-alone programs.	1. Applets are not stand-alone programs.
2. The main() method exists.	2. There is no main() method.
3. They need a Java interpreter for execution.	3. They need a browser like netscape for execution.
4. They have no hard disk accessing restrictions.	4. They can't access hard disk.
5. They doesn't need any security.	5. They need topmost security for hard disk files.

Q9. Write about tabbed pane.**Answer :**

A tabbed pane is a component which appears as a group of folders in a file cabinet. Every folder has a title. The contents become visible when a folder is selected. Only one folder can be selected at a time. Tabbed panes are encapsulated by the JTappedPane class, which extends JComponent. The below example shows how we can create a tabbed pane.

Model Paper-III, Q1(i)

Q10. Define JDialog.**Answer :**

JDialog is a type of swing class that is used to create a dialog. It does not inherit Jcomponent. It is a top-level container which is heavy in its weight than others. JOptionPane makes use of JDialog for creating dialogs. It has similar features to that of JFrame and it can be created and managed as a JFrame is done. It is inherited from AWT classes such as container, component, window and dialog.