Unit 1: The Applet Class

1. **Define applet in java.**

   Ans:

   Applets are small Java applications that can be accessed on an Internet server, transported over Internet, and can be automatically installed and run as apart of a web document. Any applet in Java is a class that extends the **java.applet.Applet** class.

2. **Give some advantages of java applet.**

   Ans:

   - Very less response time as it works on the client side.
   - Can be run using any browser, which has JVM running in it.

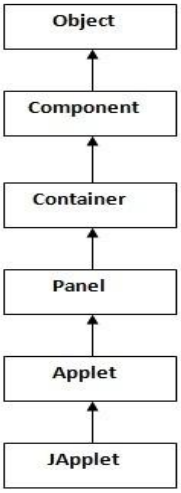
3. **How to run an Applet?**

   Ans:

   There are two ways to run an applet
   1. By html file.
   2. By appletViewer tool (for testing purpose).

4. **Show the Hierarchy of java applet.**

   Ans:



5. **Who is responsible to manage the life cycle of an applet?**

   Ans:

   Java Plug-in software is responsible to manage the life cycle of applet.

6. **How to pass parameter in an applet?**

   Ans:

   public String getParameter(String parameterName)

7. **Define HTML applet tag.**

   Ans:

The HTML <applet> tag specifies an applet. It is used for embedding a Java applet within an HTML document. It is not supported in HTML5.

8. **What are the types of applet?**

   Ans:

   There are two types of applets, they are:

   - Local Applets
   - Remote Applets

9. **Define local applets.**

   Ans:

   The local applets are applet that are developed and stored in local system. The web page will search local system directories, find the local applet and execute it. Execution of local applet does not require internet connection.

10. **Define remote applets.**

    Ans:

    The remote applets are applet types that are developed and stored in remote computer. The web page requires internet connection to locate and load the remote applet from the remote computer.

Unit 2: Event handling

1. **Define event handling.**

   Ans:

   Any program that uses GUI (graphical user interface) such as Java application written for windows, is event driven. Event describes the change of state of any object.

   Example : Pressing a button, Entering a character in Textbox.

2. **Briefly define the component of event handling.**

   Ans:

   Event handling has three main components.

   - **Events :** An event is a change of state of an object.
   - **Events Source :** Event source is an object that generates an event.
   - **Listeners :** A listener is an object that listens to the event. A listener gets notified when an event occurs.

3. **How events are handled?**

   Ans:

   A source generates an Event and send it to one or more listeners registered with the source. Once event is received by the listener, they processes the event and then return. Events are supported by a number of Java packages, like **java.util**, **java.awt** and **java.awt.event**.

4. **Point out some of the interface in java event handling.**

   Ans:

- ActionListener
- AdjustmentListener
- ComponentListener
- FocusListener
- ItemListener

5. Define JcheckBox.

Ans:

Classes JcheckBox and JradioButton are subclasses of JtoggleButton. A JradioButton is different from a JcheckBox in that normally several JradioButton are grouped together and are mutually exclusive only one in the group can be selected at any time where as more than one check box can be selected at a time.

6. Define Listener Interface.

Ans:

An event listener interface defines the methods used by a component to dispatch events. Each event type will have at least one corresponding dispatch method in a listener interface.

7. Define Adapter classes.

Ans:

An adapter class provides the default implementation of all methods in an event listener interface. Adapter classes are very useful when you want to process only few of the events that are handled by a particular event listener interface. You can define a new class by extending one of the adapter classes and implement only those events relevant to you.

8. Define Inner Class in java.

Ans:

Inner classes are a security mechanism in Java. We know a class cannot be associated with the access modifier private, but if we have the class as a member of other class, then the inner class can be made private. And this is also used to access the private members of a class.

9. Define Anonymous inner class.

Ans:

An inner class declared without a class name is known as an anonymous inner class. In case of anonymous inner classes, we declare and instantiate them at the same time.

10. Write syntax of Anonymous inner class

Ans:

```
AnonymousInner an_inner= new AnonymousInner(){
public void my_method(){
 ........
 ........
 }
};
```

Unit 3: Introduction to AWT

1. **Define AWT.**

   Ans:

   Java AWT components are platform-dependent i.e. components are displayed according to the view of operating system. AWT is heavyweight i.e. its components uses the resources of system.

   The java.awt package provides classes for AWT api such as TextField, Label, TextArea, RadioButton, CheckBox, Choice, List etc.

2. **List the window fundamentals included in AWT.**

   Ans:

   - Component
   - Container
   - Panel
   - Window
   - Frame
   - Canvas

3. **Write syntax to declare frame class in awt.**

   Ans:

   Following is the declaration for **java.awt.Frame** class:

```
public class Frame
 extends Window
implements MenuContainer
```

4. **How is JFrame size in java AWT.**

   Ans:

```
jframe.setPreferredSize(new Dimension(400, 300));
```

5. **What is difference between Swing and AWT in Java?**

   Ans:

   Swing is a considered as light weight and AWT is considered as heavy weight. Another difference between AWT and Swing is that, Swing offers uniform look and feel across platform while look and feel of AWT GUI application are platform dependent because AWT mostly use native components e.g. a AWT windows will look different in DOS and Windows operating system.

6. **What is the difference between a Scrollbar and a Scrollpane?**

   Ans:

   A Scrollbar is a Component, but not a Container. A Scrollpane is a Container and handles its own events and performs its own scrolling.

7. **What are the subclasses of the Container class?**

   Ans:

   - The Container class has three major subclasses. They are:

- Window
- Panel
- ScrollPane

8. **How can we create a borderless window?**
   Ans:
   Create an instance of the Window class, give it a size, and show it on the screen.

```
Frame aFrame = new Frame();
Window aWindow = new Window(aFrame);
aWindow.setLayout(new FlowLayout());
aWindow.add(new Button("Press Me"));
aWindow.getBounds(50,50,200,200);
aWindow.show();
```

9. **What is the difference between a MenuItem and a CheckboxMenuItem?**
   Ans:
   CheckboxMenuItem class extends the MenuItem class to support a menu item that may be checked or unchecked.

10. **Which method of the component class is used to set the position and the size of a component?**
    Ans:
    setBounds(). The following code snippet explains this:

```
txtName.setBounds(x,y,width,height);
```

UNIT: 4 Using AWT Controls, Layout Mangers, and Menus

1. Which method do you use to add a menu bar to a top-level container such as a JFrame?

To add a menu bar to a top-level conatiner such as Jframe setJMenuBar method is used.

2. Make a copy of MyDemo1.java named MyDemo2.java. Add a menu bar to MyDemo2.

```
JMenu menu = new JMenu("Menu");
JMenuBar mb = new JMenuBar();
mb.add(menu);
frame.setJMenuBar(mb);
```

3. Which abstract class is the super class of all menu related classes.

Menu Component

4. Which class can be used to represent a checkbox with a textual label that can appear in a menu?

CheckBoxMenuItem can be used to represent a checkbox with a textual label that can appear in a menu.

5. What is meant by Controls and what are different types of controls?

Controls are componenets that allow a user to interact with your application. The AWT supports the following types of controls:

i. Labels
ii. Push buttons
iii. Check boxes
iv. Choice lists
v. Lists
vi. Scroll bars
vii. Text components

6. What is the difference between a Scollbar and a Scrollpane?

A Scrollbar is a Component, but not a Container. A Scrollpane is a Container and handles its own events and performs its own scrolling.

7. What methods are used to get and set the text label displayed by a Button object?

getLabel( ) and setLabel( )

8. How can the Checkbox class be used to create a radio button?

By associating Checkbox objects with a CheckboxGroup.

9. What are the subclasses of the Container class?

The Container class has three major subclasses. They are:

i. Window
ii. Panel
iii. ScrollPane

10. How can we create a borderless window?

Create an instance of the Window class, give it a size, and show it on the screen.

```
Frame aFrame = new Frame();

Window aWindow = new Window(aFrame);

aWindow.setLayout(new FlowLayout());

aWindow.add(new Button("Press Me"));

aWindow.getBounds(50,50,200,200);

aWindow.show();
```

UNIT: 5 Introducing Swing

1. **Define SWING.**
   Swing API is a set of extensible GUI Components to ease developer's life to create JAVA based Front End/GUI Applications.

2. **What do you mean by container?**

   Containers are integral part of SWING GUI components. A container provides a space where a component can be located. A Container in AWT is a component itself and it adds the capability to add component to itself.

3. **List the two key swing features.**

   The two key features of swing are:
   i. Light weight
   ii. Pluggable look-and-feel

4. **Swing is built on the AWT expalin**.

   Swing has almost every control corresponding to AWT controls and acts as a replacement of AWT API, so swing is built on the AWT.

5. **List some GUI swing components**.

   Some GUI swing components are:
   i. Jlabel
   ii. Jbutton
   iii. JcolorChooser

iv. Jlist
v. JtextArea

6. **Give some examples of GUI based Applications**.

Some of the examples are:

i. Automated Teller Machine (ATM)
ii. Airline Ticketing Machine
iii. Mobile Applications
iv. Navigation Systems

7. **Why is Swing light weighted**?

Swing components are independent of native Operating System's API as Swing API controls are rendered mostly using pure JAVA code instead of underlying operating system calls.

8. **What is Container & Component in Java**?

A container is the abstract base class for the non menu user-interface controls of SWING. Component represents an object with graphical representation. A container is a component that can contain other SWING components.

9. What is the difference between Swing and AWT?

Swing is a considered as light weight and AWT is considered as heavy weight. Another difference between AWT and Swing is that, Swing offers uniform look and feel across platform while look and feel of AWT GUI application are platform dependent because AWT mostly use native components e.g. a AWT windows will look different in DOS and Windows operating system.

10. List some advantages of GUI over CUI.

Some of the advantages are:

i. GUI makes the application more entertaining and interesting on the other hand CUI does not.
ii. GUI offers click and execute environment while in CUI every time we have to enter the command for a task.

UNIT: 6 Exploring Swing

1. **Find a component that displays an icon, but that doesn't react to user clicks.**

   A component that displays an icon, but doesn't react to user clicks is Jlabel.

2. **Which Swing components use ListSelectionModel?**

   JList and JLabel

3. **Create a JLabel object.**

   JLabel label = new JLabel("This is a basic label");

4. **What are the functions of JButton and JList do?**

   JButton creates a labeled button whereas JList presents the user with a scrolling list of text items.

5. **What do you mean by JRadioButton?**

   The class JRadioButton is an implementation of a radio button - an item that can be selected or deselected, and which displays its state to the user.

Class Declaration:

```
public class JRadioButton
  extends JToggleButton
    implements Accessible
```

6. Create JTextField.

```
public class JTextField
  extends JTextComponent
    implements SwingConstants
```

7. What does ImageIcon do?

ImageIcon control is an implementation of the Icon interface that paints Icons from Images.

8. How to create checkbox?

```
public JCheckBox()
JCheckBox aCheckBox = new JCheckBox();
```

9. Explain the function of JOptionPane.

JOptionPane provides set of standard dialog boxes that prompt users for a value or informs them of something.

10. What is the use of JComboBox?

A JComboBox component presents the user with a to show up menu of choices.

Unit 7: Enterprise Application Architecture

1. TDD stands for:
    - Answer: Test Driven Design or Test Driven Development

2. One object oriented design technique is:
    - Answer: a. Noun Harvesting
    - Note:Noun harvesting is a classic OOD technique

3. A UML Class Diagram does not sufficiently describe an OO system because:
    - Answer:d. It is a static diagram
    - Note:Class diagrams do not document object interactions
    - Note:Other diagrams like sequence and activity diagrams capture dynamic interactions

4. The difference between aggregation and composition is:
    - Answer: Composition is an "owning" relationship
    - Note:In composition, if an object is removed, it's related objects are also removed
    - Note:In aggregations, related objects can exist on their own

5. A domain model is:
    - Answer: c. An object oriented model of the business that the application is automating
    - Note:The term "domain" refers to the business domain being automated by the application

6. The difference between "rich" and "anemic" domain models is:
    - Answer: a. A rich model has business logic

7. Refactoring is:
    - Answer: c. A re-architecture of the code that does not change it's behavior

8. Agile methods are:
    - Answer: d. Considered a current industry best practice

9. What is the most significant cause of concern for Java Annotations?
    - Answer: C: The represent a tight coupling
    - Note: Annotations represent a tight coupling to a specific framework.
10. Regarding CMP and JPA:
    - Answer: d. JPA replaces CMP

Unit 8: JDBC

1. What is JDBC?

JDBC stands for Java Database Connectivity, which is a standard Java API for database-independent connectivity between the Java programming language and a wide range of databases.

2. Describe a general JDBC Architecture.

General JDBC Architecture consists of two layers JDBC API (This provides the application-to-JDBC Manager connection) and JDBC Driver API (This supports the JDBC Manager-to-Driver Connection).

3. What are the common JDBC API components?

JDBC API consists of following interfaces and classes DriverManager, Driver, Connection, Statement, ResultSet, SQLException.

4. What is a JDBC DriverManager?

JDBC DriverManager is a class that manages a list of database drivers. It matches connection requests from the java application with the proper database driver using communication subprotocol.

5. What is a JDBC Driver?

JDBC driver is an interface enabling a Java application to interact with a database. To connect with individual databases, JDBC requires drivers for each database. The JDBC driver gives out the connection to the database and implements the protocol for transferring the query and result between client and database.

6. What is a connection?

Connection interface consists of methods for contacting a database. The connection object represents communication context.

7. What is a statement?

Statement encapsulates an SQL statement which is passed to the database to be parsed, compiled, planned and executed.

8. What is a ResultSet?

These objects hold data retrieved from a database after you execute an SQL query using Statement objects. It acts as an iterator to allow you to move through its data. The java.sql.ResultSet interface represents the result set of a database query.

9. What are the different types of JDBC Statements?

Types of statements are

- **Statement** − regular SQL statement.
- **PreparedStatement** − more efficient than statement due to pre-compilation of SQL.
- **CallableStatement** − to call stored procedures on the database.

10. What do you mean by fastest type of JDBC driver?

JDBC driver performance or fastness depends on a number of issues Quality of the driver code, size of the driver code, database server and its load, Network topology, Number of times your request is translated to a different API.

11. Is there a practical limit for the number of SQL statements that can be added to an instance of a Statement object?

The specification makes no mention of any size limitation for Statement.addBatch(), this is dependent, on the driver.

12. How do you update a result set?

ResultSet interface contains a collection of update methods for updating the data of a result set. Each update method has two versions for each data type

- One that takes in a column name.
- One that takes in a column index.

These methods change the columns of the current row in the ResultSet object, but not in the underlying database. To update your changes to the row in the database, you need to invoke one of the following methods

updateRow(), deleteRow(), refreshRow(), cancelRowUpdates(), insertRow()

13. How does JDBC handle the data types of Java and database?

The JDBC driver converts the Java data type to the appropriate JDBC type before sending it to the database. It uses a default mapping for most data types. For example, a Java int is converted to an SQL INTEGER.

14. Why would you use a batch process?

Batch Processing allows you to group related SQL statements into a batch and submit them with one call to the database.

15. What is a transaction?

A transaction is a logical unit of work. To complete a logical unit of work, several actions may need to be taken against a database. Transactions are used to provide data integrity, correct application semantics, and a consistent view of data during concurrent access.

16. What is a RowSet?

A JDBC RowSet object holds tabular data in a way that makes it more flexible and easier to use than a result set. A RowSet objects are JavaBeans components.

17. What is Metadata and why should you use it?

JDBC API has two Metadata interfaces DatabaseMetaData&ResultSetMetaData. The meta data provides comprehensive information about the database as a whole. The implementation for these interfaces is implemented by database driver vendors to let users know the capabilities of a Database.

Unit 9: XML and JAVA

1. What XML stands for?

XML stands for Extensible Markup Language.

What are the advantages of using XML?

Following are the advantages that XML provides −

- **Technology agnostic** - Being plain text, XML is technology independent. It can be used by any technology for data storage and transmission purpose.
- **Human readable**- XML uses simple text format. It is human readable and understandable.
- **Extensible** - in XML, custom tags can be created and used very easily.
- **Allow Validation** - Using XSD, DTD and XML structure can be validated easily.

2. What are the disadvantages of using XML?

Following are the disadvantages of XML usage −

- **Redundant Syntax** - Normally XML file contains lot of repeatitive terms.
- **Verbose**-Being a verbose language, XML file size increases the transmission and storage costs.

3. What is XML Parsing?

Parsing XML refers to going through XML document to access data or to modify data in one or other way.

4. What is XML Parser?

XML Parser provides way how to access or modify data present in an XML document. Java provides multiple options to parse XML document.

5. What DOM stands for?

DOM stands for Document Object Model.

6. What is DOM?

DOM stands for Document Object Model and it is an official recommendation of the World Wide Web Consortium (W3C). It defines an interface that enables programs to access and update the style, structure,and contents of XML documents. XML parsers that support the DOM implement that interface.

7. When to use a DOM Parser?

You should use a DOM parser when −

- You need to know a lot about the structure of a document
- You need to move parts of the document around (you might want to sort certain elements, for example)
- You need to use the information in the document more than once

8. What DOM Parser returns?

When you parse an XML document with a DOM parser, you get back a tree structure that contains all of the elements of your document. The DOM provides a variety of functions you can use to examine the contents and structure of the document.

9. What are the advantages of DOM Parsing?

The DOM is a common interface for manipulating document structures. One of its design goals is that Java code written for one DOM-compliant parser should run on any other DOM-compliant parser without changes.

10. Can we create an XML document using DOM parser?

Yes! Using DOM parser, we can parse, modify or create a XML document.

11. What SAX stands for?

SAX stands for Simple API for XML.

12. What is a SAX Parser?

SAX Parser is an event-based parser for xml documents.

13. What are the disadvantages of SAX Parsing?

- We have no random access to an XML document since it is processed in a forward-only manner
- If you need to keep track of data the parser has seen or change the order of items, you must write the code and store the data on your own

14. Name some methods for processing the attributes connected to an element in SAX parsing.

Attributes Interface specifies methods for processing the attributes connected to an element.

- **intgetLength()** - Returns number of attributes.
- **String getQName(int index)**
- **String getValue(int index)**
- **String getValue(String qname)**

15. Can we create an XML document using SAX parser?

No! Using SAX parser, we can only parse or modify a XML document.

16. What is JDOM Parser?

JDOM is an open source, java based library to parse XML document and it is typically java developer friendly API.

17. What are the benefits of JDOM parser?

It is java optimized, it uses java collection like List and Arrays. It works with DOM and SAX APIs and combines the best of the two. It is of low memory footprint and is nearly as fast as SAX.

18. Can we create an XML document using JDOM parser?

Yes! Using JDOM parser, we can parse, modify and create a XML document.

19. Can we create an XML document using StAX parser?

Yes! Using StAX parser, we can parse, modify and create a XML document.

20. Explain XMLEventReader Class of StAX parser.

This class provide iterator of events which can be used to iterate over events as they occur while parsing the XML document.

- **StartElementasStartElement()** - used to retrieve value and attributes of element.
- **EndElementasEndElement()** - called at the end of a element.
- **Characters asCharacters()** - can be used to obtain characters such a CDATA, whitespace etc.

21. Explain XMLEventWriter Class of StAX parser.

This interface specifies methods for creating an event.

- **add(Event event)** - Add event containing elements to XML.

22. Explain XMLStreamReader Class of StAX parser.

This class provide iterator of events which can be used to iterate over events as they occur while parsing the XML document

- **intnext()** - used to retrieve next event.
- **booleanhasNext()** - used to check further events exists or not
- **String getText()** - used to get text of an element
- **String getLocalName()** - used to get name of an element

23. Explain XMLStreamWriter Class of StAX parser.

This interface specifies methods for creating an event.

- **writeStartElement(String localName)** - Add start element of given name.
- **writeEndElement(String localName)** - Add end element of given name.
- **writeAttribute(String localName, String value)** - Write attribute to an element.

Some questions and answers on AWT: CONTROLS, LAYOUT MANAGERS AND MENUS:

1) What is meant by Controls and what are different types of controls?
Controls are components that allow a user to interact with your application.
The AWT supports the following types of controls:
Labels
Push buttons
Check boxes
Choice lists
Lists
Scroll bars
Text components
These controls are subclasses of Component.

2) You want to construct a text area that is 80 character-widths wide and 10 character-heights tall. What code do you use?
new TextArea(10, 80)

3) A text field has a variable-width font. It is constructed by calling new TextField("iiiii"). What happens if you change the contents of the text field to "wwwww"? (Bear in mind that i is one of the narrowest characters, and w is one of the widest.)
The text field stays the same width; to see the entire contents you will have to scroll by using the ß and à keys.

4) The CheckboxGroup class is a subclass of the Component class.
False

5) What are the immediate super classes of the following classes?
a) Container class
b) MenuComponent class
c) Dialog class
d) Applet class
e) Menu class

Ans: a) Container - Component
b) MenuComponent - Object
c) Dialog - Window
d) Applet - Panel
e) Menu - MenuItem

6) What are the SubClass of Textcomponent Class?
TextField and TextArea

7) Which method of the component class is used to set the position and the size of a component?
setBounds()

8) Which TextComponent method is used to set a TextComponent to the read-only state?
setEditable()

9) How can the Checkbox class be used to create a radio button?
By associating Checkbox objects with a CheckboxGroup.

10) What Checkbox method allows you to tell if a Checkbox is checked?
getState()

11) Which Component method is used to access a component's immediate Container?
getParent()

12) What methods are used to get and set the text label displayed by a Button object?
getLabel( ) and setLabel( )

13) What is the difference between a Choice and a List?
A Choice is displayed in a compact form that requires you to pull it down to see the list of available choices. Only one item may be selected from a Choice.

A List may be displayed in such a way that several List items are visible. A List supports the selection of one or more List items.

14) Which Container method is used to cause a container to be laid out and redisplayed?
validate( )

15) What is the difference between a Scollbar and a Scrollpane?
A Scrollbar is a Component, but not a Container.
A Scrollpane is a Container and handles its own events and performs its own scrolling.

16) Which Component subclass is used for drawing and painting?
Canvas.

17) Which of the following are direct or indirect subclasses of Component?
a) Button
b) Label
c) CheckboxMenuItem
d) Toolbar
e) Frame
Ans: a, b and e

18) Which of the following are direct or indirect subclasses of Container?
a) Frame
b) TextArea
c) MenuBar
d) FileDialog
e) Applet
Ans: a,d and e

19) Which method is used to set the text of a Label object?
a) setText( )
b) setLabel( )
c) setTextLabel( )
d) setLabelText( )
Ans: a

20) Which constructor creates a TextArea with 10 rows and 20 columns?
a) new TextArea(10, 20)
b) new TextArea(20, 10)
c) new TextArea(new Rows(10), new columns(20))
d) new TextArea(200)
Ans: a [Usage is TextArea(rows, columns)]

21) Which of the following creates a List with 5 visible items and multiple selection enabled?
a) new List(5, true)
b) new List(true, 5)
c) new List(5, false)
d) new List(false,5)

Ans: a [Usage is List(rows, multipleMode)]

22) Which are true about the Container class?
a) The validate( ) method is used to cause a Container to be laid out and redisplayed.
b) The add( ) method is used to add a Component to a Container.
c) The getBorder( ) method returns information about a Container's insets.
d) The getComponent( ) method is used to access a Component that is contained in a Container.
Ans: a, b and d

23) Suppose a Panel is added to a Frame and a Button is added to the Panel. If the Frame's font is set to 12-point TimesRoman, the Panel's font is set to 10-point TimesRoman, and the Button's font is not set, what font will be used to display the Button's label?
a) 12-point TimesRoman
b) 11-point TimesRoman
c) 10-point TimesRoman
d) 9-point TimesRoman
Ans: c

24) A Frame's background color is set to Color. Yellow, and a Button's background color is to Color.Blue. Suppose the Button is added to a Panel, which is added to the Frame. What background color will be used with the Panel?
a) Color.Yellow

25) Which method will cause a Frame to be displayed?
a) show( )
b) setVisible( )
c) display( )
d) displayFrame( )
Ans: a and b

26) All the componenet classes and container classes are derived from _____ class.
Ans: Object.

27) Which method of the container class can be used to add components to a Panel.
Ans: add ( ) method.

28) What are the subclasses of the Container class?
Ans: The Container class has three major subclasses. They are :
Window
Panel
ScrollPane

29) The Choice component allows multiple selection.
a) True.
b) False.
Ans: b

30) The List component does not generate any events.
a) True.
b) False.

Ans: b

31) Which components are used to get text input from the user.
Ans: TextField and TextArea.

32) Which object is needed to group Checkboxes to make them exclusive?
Ans: CheckboxGroup.

33) Which of the following components allow multiple selections?
a) Non-exclusive Checkboxes.
b) Radio buttons.
c) Choice.
d) List.
Ans: a and d.

34) What are the types of Checkboxes and what is the difference between them?
Ans: Java supports two types of Checkboxes. They are : Exclusive and Non-exclusive.
In case of exclusive Checkboxes, only one among a group of items can be selected at a time. If an item from the group is selected, the checkbox currently checked is deselected and the new selection is highlighted. The exclusive Checkboxes are also called as Radio buttons.
The non-exclusive checkboxes are not grouped together and each one can be selected independent of the other.

35) What is a Layout Manager and what are the different Layout Managers available in java.awt and what is the default Layout manager for the panal and the panal subclasses?
Ans: A layout Manager is an object that is used to organize components in a container.
The different layouts available in java.awt are :
FlowLayout, BorderLayout, CardLayout, GridLayout and GridBag Layout.
The default Layout Manager of Panal and Panal sub classes is FlowLayout".

36) Can I exert control over the size and placement of components in my interface?
Ans: Yes.
myPanal.setLayout(null);
myPanal.setbounds(20,20,200,200);

37) Can I add the same component to more than one container?
Ans: No. Adding a component to a container automatically removes it from any previous parent(container).

38) How do I specify where a window is to be placed?
Ans: Use setBounds, setSize, or setLocation methods to implement this.
setBounds(int x, int y, int width, int height)
setBounds(Rectangle r)
setSize(int width, int height)
setSize(Dimension d)
setLocation(int x, int y)
setLocation(Point p)

39) How can we create a borderless window?
Ans: Create an instance of the Window class, give it a size, and show it on the screen.
eg. Frame aFrame = ...

```
Window aWindow = new Window(aFrame);
aWindow.setLayout(new FlowLayout());
aWindow.add(new Button("Press Me"));
aWindow.getBounds(50,50,200,200);
aWindow.show();
```

40) Can I create a non-resizable windows? If so, how?
Ans: Yes. By using setResizable() method in class Frame.

41) What is the default Layout Manager for the Window and Window subclasses (Frame,Dialog)?
Ans: BorderLayout().

42) How are the elements of different layouts organized?
Ans: FlowLayout : The elements of a FlowLayout are organized in a top to bottom, left to right fashion.
BorderLayout : The elements of a BorderLayout are organized at the
borders (North, South, East and West) and the center of a
container.
CardLayout : The elements of a CardLayout are stacked, one on top of the other, like a deck of cards.
GridLayout : The elements of a GridLayout are of equal size and are laid out using the square of a grid.
GridBagLayout : The elements of a GridBagLayout are organized according to a grid.However, the elements are of different sizes and may occupy
more than one row or column of the grid. In addition, the rows and columns may have different sizes.

43) Which containers use a BorderLayout as their default layout?
Ans: The Window, Frame and Dialog classes use a BorderLayout as their default layout.

44) Which containers use a FlowLayout as their default layout?
Ans: The Panel and the Applet classes use the FlowLayout as their default layout.

45) What is the preferred size of a component?
Ans: The preferred size of a component size that will allow the component to display normally.

46) Which method is method to set the layout of a container?
a) startLayout( )
b) initLayout( )
c) layoutContainer( )
d) setLayout( )
Ans: d

47) Which method returns the preferred size of a component?
a) getPreferredSize( )
b) getPreferred( )
c) getRequiredSize( )
d) getLayout( )
Ans: a

48) Which layout should you use to organize the components of a container in a tabular form?
a) CardLayout

b) BorederLayout
c) FlowLayout
d) GridLayout
Ans: d

49) An application has a frame that uses a Border layout manager. Why is it probably not a good idea to put a vertical scroll bar at North in the frame?
a) The scroll bar's height would be its preferred height, which is not likely to be enough.
b) The scroll bar's width would be the entire width of the frame, which would be much wider than necessary.
c) Both a and b.
d) Neither a nor b. There is no problem with the layout as described.
Ans: c

50) What is the default layouts for a applet, a frame and a panel?
Ans : For an applet and a panel, Flow layout is the default layout, whereas Border layout is default layout for a frame.
If a frame uses a Grid layout manager and does not contain any panels, then all the components within the frame are the same width and height.
a) True
b) False.
Ans: a

51) If a frame uses its default layout manager and does not contain any panels, then all the components within the frame are the same width and height.
a) True
b) False.
Ans: b

52) With a Border layout manager, the component at Center gets all the space that is left over, after the components at North and South have been considered.
a) True
b) False
Ans: b

53) An Applet has its Layout Manager set to the default of FlowLayout. What code would be the correct to change to another Layout Manager?
a) setLayoutManager(new GridLayout());
b) setLayout(new GridLayout(2,2));
c) setGridLayout(2,2,))
d setBorderLayout();
Ans: b

54) How do you indicate where a component will be positioned using Flowlayout?
a) North, South,East,West
b) Assign a row/column grid reference
c) Pass a X/Y percentage parameter to the add method
d) Do nothing, the FlowLayout will position the component
Ans:d

55) How do you change the current layout manager for a container?

a) Use the setLayout method
b) Once created you cannot change the current layout manager of a component
c) Use the setLayoutManager method
d) Use the updateLayout method
Ans :a

56) When using the GridBagLayout manager, each new component requires a new instance of the GridBagConstraints class. Is this statement true or false?
a) true
b) false
Ans: b