

Prepared by: Dr. Tariq Hanif
Midterm-13

For more help @:

Email: qirathanif@gmail.com

Website: drqirathanif.jimdo.com

Cell# 03037300008

MCGQS

1. ResultSetMetaData object can be derived from _____ object.

Connection

Statement

ResultSet (pg162)

PreparedStatement

2. Which one of the following is NOT correct declaration?

String args[]

String[]arguments

String arguments[]

String[] args[]

3. Which is the main class in JAVA from which errors derive?

Exception

IOException

Throwable

RuntimeException

4. Identify those classes that are not GUI based.

Button classes

Frame classes

Vector classes (pg 102)

RadioButton classes

5. If we want to handle mouse Movement or mouse drag. Which of the following interface do we need to implement?

MouseMotionListener (pg 130)

MouseListener

MouseDragged

MouseMoved

1

6. If a mouse clicks on "x" portion on the top of the window. Which Action event is it showing?

- Mouse Pressed
- Mouse Released

Mouse Clicked (pg 119)

Mouse Enter

7. Result set index starts from :

0 (pg 162)

Any number

-1

1

8. Which of the following method is called only once in Applets?

start()

init() (pg 267)

stop()

paint()

9. What is not true about communication socket:

It listens client's data requests.

It communicates with client.

It is ordinary socket.

It runs on client machine.

10. Java defines _____ for the listener interfaces having more than one event handling methods.

Wrapper classes

Adapter classes (pg 127)

Abstract classes

Concrete classes

11. Which of the following method is used to determine whether a database is read only or not?

isReadOnly() (pg 177)

isreadonly()

isreadOnly()

isReadOnly()

12. ServerSocket's accept() method waits for incoming connection request and when request is received, it returns a _____.

Server socket

Communication socket

Message associated with the request

Both Server socket and Message associated with the request

13. How many objects are created in the following piece of code?

2

Student s1, s2, s3, s4;

s1 = new Student ();

s3 = new Student ();

1
2
3
4

14. Consider the following code segment.

```
class CommonErrors {  
public static void main(String[] args){  
    int num = 10 ;  
    System.out.println(number);  
}  
}
```

What type of error can occur in the above code segment?

Local variable not initialized

Cannot find symbol

Null Pointer exception

No Such Method error

15. Which of the following exception belongs to a category of checked exception?

Null Pointer exception

Array index out of bounds

IOException (pg 107)

NumberFormatException

16. Which one of the following is pure java component?

Swing Components (pg 100)

AWT Components

GUI Components

Heavy weight components

17. In a Java GUI, the purpose of classes which implement the ActionListener interface is to:

Describe the graphical display of GUI items

Control the organization of items displayed in the GUI

Respond to the events caused by GUI items

Control GUI items.

18. If a class needs to handle events generated by button then which of the following interface a class needs to implement?

MouseListener

KeyListener

ActionListener (pg 122)

ComponentListener

19. The following code is an example of_____.

```
double a=123.45;
```

```
int b= (int) a;
```

Up-casting

3

Down-casting (pg 59)

Normal-casting

High-casting

20. == Operator will use _____ to compare strings.

Shallow Comparison

Deep comparison

Both shallow and deep comparison

None of the given options (pg 38)

1. If a sub class overrides all abstract methods of super class then it becomes _____
abstract class

concrete class (pg 93)

wrapper class

adapter class

2. What will happen if the static modifier is removed from the signature of the main method?

The program does not compile

The program compiles but does not run

The program compiles and runs properly

The program throws an exception on compile time

3. Which of the following 'statement' object is used for executing precompiled SQL statements?

Statement

PreparedStatement (pg 147)

CallableStatement

None of the given options

4. What will be the output from the following code?

```
System.out.print("Virtual University.");
```

```
System.out.println("Islamabad.");
```

```
System.out.println("Campus.");
```

Virtual University. Islamabad. Campus.

Virtual University. Islamabad.

Campus.

Virtual University.

Islamabad. Campus.

Virtual University.

Islamabad.

Campus.

5. If the implementation of event generator{} class and event handler{} class is done in the same class which keyword is passed in this statement for registering event with generator instead of x:

```
Display.addWindowListener(x);
```

new

this

super

sub

6. When you maximize the applet which was minimized a few times before which of the following methods will be called?

start()

stop()

init()

destroy()

7. Stream is an abstraction of hard drive same as socket is an abstraction of _____.

Network (pg 194)

file

data

memory

8. Which of the following GUI components can be created?

Mouse

Keys

Buttons (pg 120)

Both Mouse and Keys

9. Which of the following objects is passed as an argument to the paint () method?

A Canvas object

A Graphics object

An Image object

A paint object

10. Static methods can only access _____.

Instance variables

Instance methods

Static variables and methods (pg 50)

Both static and instance members

11. A _____ in HashMap is associated with each object that is stored.

Key

Value

Data member

Attribute

12. An exception in Java is represented as a/an _____.

Operator

Object (pg 70)

Function

Primitive data type

13. Which of the following is a top-level container?

Dialog (pg 101)

JPanel

ScrollPane

ToolBar

5

14. Consider the following code:

```
public class abc extends JFrame implements ActionListener {
```

```
Container contentPane = getContentPane();  
  
JButton button = new JButton("button");  
  
}
```

Which of the following correctly adds the button to the frame?

```
add(button, BorderLayout.NORTH);  
abc.add(button);  
add(button); (pg 106)  
contentPane.add(button);
```

15. Which one of the following is not supported by Java?

Single inheritance
Multiple inheritance
Data hiding
Data encapsulation

16. The "MouseDragged" and "mouseMoved" methods are defined in _____ interface.

ActionListener
ItemListener
MouseListener

MouseMotionListener (pg122)

17. A class that is defined in another class is called _____.

Private class
Public class
Inner class
Derived class

18. Which of the following statement regarding Java is false?

Java is a programming language
Java is case sensitive
Java uses an interpreter

A Java program compiles into machine language

19. _____ translates all JDBC calls into ODBC calls and send them to the ODBC Driver.

Type-1 (pg168)

Type-2
Type-3
Type-4

20. Which of the following method we should override while painting?

paintBorder() (pg 173)

paint()
paintChildren ()
paintComponent()

Subjective

21. What is the difference between mouse pressed and mouse clicked?02

Answer: clicked = pressed + released

mouse pressed is mouse pressed while mouse clicked is mouse pressed plus mouse released

22. How can we work with database having closed connections? 02

Answer: we can not work with database in java having closed connections, to work with database in java there must be a connection

23. In term of coding event handling is three steps process. Writes these steps in correct order. 03

Answer: • **Step 1:** Create components which can generate events (Event Generators)

• **Step 2:** Build component (objects) that can handle events (Event Handlers)

• **Step 3:** Register handlers with generators

24. Suppose you have given ResultSet object rs. Write the java code to get the maximum width of 2nd column in the result set. 03

Answer: ResultSetMetaData rsmd= rs.getMetaData();

String Cname= rsmd.getColumnname(2);

System.out.println("Name of second Column is : " + Cname);

PreparedStatement pStmt = con.prepareStatement(sql,

ResultSet.TYPE_SCROLL_INSENSITIVE, ResultSet.CONCUR_UPDATABLE);

21. How ItemListener interface is defined in JDK. Just write code02

Answer: public interface ItemListener {

public void itemStateChanged(ItemEvent e);

}

22. In java, how is it possible to get the name of a specific column in the ResultSet?02

Answer: The getColumnname()method returns the database name of the column.

23. What is function of getQueryTimeout and setQueryTimeout methods in java?03

Answer: • Retrieves the number of seconds the driver will wait for a Statement object to execute.

• The current query time out limit in seconds, zero means there is no limit

• If the limit is exceeded, a SQLException is thrown

24. Suppose you have given ResultSet object rs. You being a Java developer is required to write a Java code that will get the name of 2nd column in the result set.

Note:

No need to right complete program just write necessary code. 03

Answer: rs.absolute(2);

```
// update address column of 2nd row in rs
rs.updateString("Address", "model town");
// update the row in database
rs.updateRow();
//Step 8: close the connection
```

25. Create a client program that takes user's NIC number and sends it to the server whose IP address is:196.192.1.1 residing on the port 1299.05

Answer: Client side programming

create a communication socket

```
Socket s = new Socket("196.192.1.1", 1299.05);
```

asking user to enter his/her NIC

```
String msg = JOptionPane.showInputDialog("Enter your NIC");
```

sending NIC to server

```
pw.println(msg);
```

s26. Following program is used to update the student address (column Name "add" and string address need to be updated to "Model Town") in resultset and also update it in the database. Complete the following code using PreparedStatement object which is updatable and scrollable. Write only the required portion of code without rewriting the whole program.

```
// File UpdateStdRS.java
```

```
import java.sql.*;
```

```
public class UpdateStdRS
```

```
{
```

```
public static void main (String args[ ])
```

```
{
```

```
try
```

```
{
```

```
string empAdress ="Islamabad";
```

```
int empID =12;
```

```
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
```

```
String url = "jdbc:odbc:personDSN";
```

```
Connection con = DriverManager.getConnection(url, "", "");
```

```
String sql = "SELECT * from student";
```

```
//write your code here
```

```
con.close();
```

```
}catch(Exception sqlEx)
```

```
{
```

```
System.out.println(sqlEx);
```

```
}
```

```
} // end main
```

```
} // end class
```

Note: any change in the existing code will not be accepted.

05

Answer:

```
// File UpdateStdRS.java
import java.sql.*;
public class UpdateStdRS
{
public static void main (String args[ ])
{
try
{
    string empAdress ="Islamabad";
    int empID =12;
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    String url = "jdbc:odbc:personDSN";
    Connection con = DriverManager.getConnection(url, "", "");
    String sql = "SELECT * from student";
    PreparedStatement pStmt =
    con.prepareStatement(sql,ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CO
    NCUR_UPDATABLE);
    ResultSet rs =pStmt. excuteQuery();

    con.close();
}catch(Exception sqlEx)
{
System.out.println(sqlEx);
}
} // end main
} // end class
```

1. **Quiz: What are advantages of updateable/scrollable ResultSet on default ResultSet?**

Write two only.

Ans: It is possible to produce ResultSet objects that are scrollable and/or updatable. • With the help of such ResultSet, it is possible to move forward as well as backward with in ResultSet object. • Another advantage is, rows can be inserted, updated or deleted by using updatable ResultSet object.

2. **Quiz: In java, how is it possible to get number of columns in ResultSet?**

Ans: **getColumnCount ()** • Returns the number of columns in the result set

getColumnDisplaySize (int) • Returns the maximum width of the specified column in characters

getColumnName(int) / getColumnLabel (int)• The **getColumnName()** method returns the database name of the column• The **getColumnLabel()** method returns the suggested column label for printouts.

9

3. Quiz: Why type-4 drivers are more efficient than other drivers? Why called thin drivers?

Ans: JDBC Driver Types

- JDBC Driver Types are divided into four types or levels.
- Each type defines a JDBC driver implementation with increasingly higher level of Platform independence, performance, deployment and administration.
- The four types are:
 - o Type - 1: JDBC - ODBC Bridge
 - o Type 2: Native - API/partly Java driver
 - o Type 3: Net - protocol/all-Java driver
 - o Type 4: Native - protocol/all-Java driver

1: JDBC - ODBC Bridge

- Translates all JDBC calls into ODBC (Open Database Connectivity) calls and send them to the ODBC Driver • Generally used for Microsoft database. • Performance is degraded.

2: Native - API/partly Java driver

- Converts JDBC calls into database-specific calls such as SQL Server, Informix, Oracle or Sybase.
- Partly-Java drivers communicate with database-specific API (which may be in C/C++) using the Java Native Interface. • Significantly better Performance than the JDBC-ODBC Bridge.

3: Net - protocol/all-Java driver

- Follows a three-tiered approach whereby the JDBC database requests () are passed through the network to the middle-tier server • Pure Java client to server drivers which send requests that are not database specific to a server that translates them into a database-specific protocol. • If the middle-tier server is written in java, it can use a type 1 or type 2 JDBC driver to do this.

4: Native - protocol / all - java driver

- Converts JDBC calls into the vendor-specific DBMS protocol so that client application can communicate directly with the database server • completely implemented in Java to achieve platform independence and eliminate deployment issues. • Performance is typically very good.

4. Question No 4 MARKS 3 Methods (paint Border, paint Children, paint Component) write correct order of methods call in if these methods are called.

Ans: paintComponent() • It is a main method for painting • By default, it first paints the background • After that, it performs custom painting (drawing circle, rectangles etc.)

2 paintBorder() • Tells the components border (if any) to paint. • It is suggested that you do not override or invoke this method

3 paintChildren()

• Tells any components contained by this component to paint themselves • It is suggested that you do not override or invoke this method too.

10 5. Question No 6 MARKS 5: Operator Text field and answer JLabels are already created?

Ans: Panel pButtons; JTextField tfAnswer; JLabel lMyCalc;

6. //method used for setting layout of GUI

7. QUIZ::what problem you can face using adaptor(3)

```

Ans: 1. import java.awt.*;
2. import javax.swing.*;
3. import java.awt.event.*;
4. public class EventsEx extends WindowAdapter {
5. JFrame frame;
6. JLabel coordinates;
// setting layout
7. public void initGUI () {
// creating event generator
8. frame = new JFrame();
9. Container cont = frame.getContentPane();
10. cont.setLayout(new BorderLayout() );

```

1. What's the start do in applet cycle and when we use it

Answer: To start the applet's execution • For example, when the applet's loaded or when the user revisits a page that contains the Applet. • start() is also called whenever the browser is maximized.

2. Write the syntax to load the oracle driver

Answer: Class.forName("oracle.jdbc.driver.OracleDriver");

3. What dose codes do?

```

ResultSet rs=TYPING_SCROLL_INSENSITIVE
ResultSet.TYPE_SCROLL_INSENSITIVE,

```

Answer: The following code fragment, illustrates how to make a ResultSet object that is scrollable and Updatable. String sql = "SELECT * FROM Person"; PreparedStatement pStmt = con.prepareStatement(sql, ResultSet.TYPE_SCROLL_INSENSITIVE, ResultSet.CONCUR_UPDATABLE);
ResultSet rs = pStmt.executeQuery();

Two constants have been used of ResultSet class for producing a ResultSet rs that is scrollable, will not show changes made by others and will be updatable.

What DO MaxRows (int) & setMaxRows(int) methods?

Answer: • Used for determines the number of rows a ResultSet may contain • By default, the number of rows are unlimited (return value is 0), or by using setMaxRows(int), the number of rows can be specified.

Write any five GUI based component.

Answer: 1. BUTTON 2. Label 3. Text filed 4. Text area 5.manue

Define paint() in applet life cycle?

Ans:• Paint () is called for the first time when the applet becomes visible • Whenever applet needs to be repainted, paint() is called again • Do all your painting in paint (), or in a method that is called from paint()

What is and why we use: JDBC - ODBC Bridge?

Answer: • Translates all JDBC calls into ODBC (Open Database Connectivity) calls and send them to the ODBC Driver • Generally used for Microsoft database. • Performance is degraded.

Q: What is the difference b/w socket and server socket, also give java syntax of socket

and ss?

Ans: A socket is one endpoint of a two-way communication link between two programs running generally on a network. Syntax for socket `Socket s = new Socket("serverName", serverPort)`; The server listens through a special kind of socket, which is named as server socket.

The sole purpose of the server socket is to listen for incoming request; it is not used for Communication Syntax for server socket: `ServerSocket ss = new ServerSocket(serverPort)`;

Q: What is disadvantage of interfaces?

Answer: One disadvantage of interfaces is that once you publish them to other coders outside your own control, they represent a public commitment which can be difficult to change later. You don't know who else is using the interface and you don't know how, so you can't just add a new

method to the interface without risking breaking the code of existing clients (if/when they upgrade). In comparison, if you'd used an abstract or concrete class, you could just add a new

concrete method with no problems (usually).

10. What problem we faced in interpreter and how do we handle that situation ?

Ans: `Public static void main(String[] args)` The java interpreter must find this defined exactly as given or it will refuse to run the program. (However you can change the name of parameter that is passed to main. i.e. you can write `String[] argv` or `String[] someParam` instead of `String[] args`)

Q: Write coding for while loop in ResultSet6. (columns) method a table was given [5 marks]

Answer: For example, the following code snippet will iterate over the whole `ResultSet` and illustrates the usage of getters methods while (`rs.next()`){`String name = rs.getString("columnName"); //by using column name` `String name = rs.getString(1); // or by using column index` }

Q: What is the process of get metadata in java language?(2)

Answer: `ResultSetMetaData rsmd = rs.getMetaData();` Now, `rsmd` can be used to look up number, names & types of columns.

Q: If we use scroll /update and same time use previous() exception throws reason?(2)

Answer: It is possible to produce `ResultSet` objects that are scrollable and/or updatable.

- With the help of such `ResultSet`, it is possible to move forward as well as backward with in `ResultSet` object `Previous()`
- Moves the cursor to the previous row in the `ResultSet` object, if available
- Returns true if cursor is on a valid row, false if it is off the result set.
- Throws exception if result type is `TYPE_FORWARD_ONLY`

Q: Which methods are used to execute queries and when they are used?(3)

Answer: Two methods are generally used for executing SQL queries. These are:

o executeQuery(sql) method Used for SQL `SELECT` queries. Returns the `ResultSet` object that contains the results of the query and can be used to access the query results.

`String sql = "SELECT * from sometable";` `ResultSet rs = stmt.executeQuery(sql);`

o executeUpdate(sql) method

This method is used for executing an update statement like `INSERT`,

☑ UPDATE or DELETE

☑ f Returns an Integer value representing the number of rows updated

☑ String sql = "INSERT INTO tablename " + "(columnNames) Values (values)" ;

☑ int count = stmt.executeUpdate(sql);

Q: Write three functions of paintComponent() method.(3)

Answer: Three methods are at the heart of painting a swing component like JPanel etc. For instance, paint() gets called when it's time to render -- then Swing further factors the paint() call into three separate methods, which are invoked in the following order:

protected void paintComponent(Graphics g), protected void paintBorder(Graphics g)

protected void paintChildren(Graphics g)

Q: Write a GUI program in which on mouse click a message comes "mouse is clicked" don't

implement interface assume class is adapter class.(5)

Answer: public class MouseMotionAdapter implements

MouseListener{

public void mouseClicked (MouseEvent me) {

JOptionPane.showMessageDialog(null, "this");}

Q. Difference b/w mouse dragged and mouse move.

Answer: public interface MouseMotionListener { public void mouseDragged (MouseEvent me);

public void mouseMoved(MouseEvent me); }

Q. Why action listener have never adapter class?

Answer: Because action listener are used with interfaces and adapter class is not used with interface. For listener interfaces containing more than one event handling methods, jdk defines

adapter classes. Examples are o For WindowListener →WindowAdapter o For MouseMotionListener →MouseMotionAdapter o and many more • Adapter classes provide definitions for all the methods (empty bodies) of their corresponding Listener interface

• It means that WindowAdapter class implements WindowListener interface and provide the

definition of all methods inside that Listener interface

22. Why this statement is called connection url? Explain by relating it to browser url? "jdbc:odbc:person DSN"

Answer: • To get a connection, we need to specify the URL of a database (Actually we need to specify the address of the database which is in the form of URL) • As we are using Microsoft Access database and we have loaded a JDBC-ODBC driver. Using JDBC-ODBC driver requires a DSN which we have created earlier and named it personDSN. So the URL of the database will be

String conURL = "jdbc:odbc:personDSN";

Q: When we call metadata function on "con" object what does it do (3)

Answer: From a Connection object, a DataBaseMetaData object can be derived. The following code snippet demonstrates how to get DataBaseMetaDataobject.

Connection con= DriverManager.getConnection(url, usr, pwd);

DatabaseMetaData dbMetaData = con.getMetaData();

Q: Can an anonyms' class solve our problem of multiple inheritances (2)

Answer: Yes it can at some extent, as it has same characteristics of inner class and inner class is a

Solution of single inheritance and allow multiple inheritances.

Q: Write the code of given query . Change the name of student NaeemKhan where its id is 10. 2 marks note: table name is Student. column name is Std_name column name of id is std_id.

Answer: String sql = "INSERT INTO Student where std_id= 10 " + "(std_name) Values ("Naeemkhan")"; int count = stmt.executeUpdate(sql);

Differentiation between Named vs. Anonymous object with example code? 3 marks

Answer: Named • String s = "hello"; System.out.println(s); • "hello" has a named reference s.

Anonymous • System.out.println("hello");

We generally use anonymous object when there is just a onetime use of a particular object but

in case of a repeated use we generally used named objects and use that named reference to use that objects again and again.

Q: Write a given connection URL in string variable conURL. How will you connect to MS Access

database using conURL in your java program?

Answer: For MS Access, load following driver available with j2se.

• Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

String conURL = "jdbc:odbc:personDSN";

Q: Complete syntax of paint Border method

Answer: protected void paintBorder(Graphics g)

Write java code that will update name Ali to Ahmed on 5th row. Consider the column name is

sname note write only necessary code?

Answer: String sql = "INSERT INTO tablename where rowno= 5 " + "(sname) Values ("ahmed")";

int count = stmt.executeUpdate(sql);

Q. One advantage and disadvantage of result set rather than normal database 3 mrk

Ans: A default ResultSet object is not updatable and has a cursor that moves forward only.

• You can iterate over through it only once and only from the first row to last row. It is possible to produce ResultSet objects that are scrollable and/or updatable. • With the help of such ResultSet, it is possible to move forward as well as backward with in ResultSet object. Another advantage is, rows can be inserted, updated or deleted by using updatable ResultSet object.

Q: What is the disadvantage of interface in implementation? 2mrks

Answer: We will have to provide definitions of all the methods because class implementing an interface has to provide definitions of all methods present in that interface.

Q: How it is different to one server and multiple clients that one to one client server.

Answer: The server needs a new socket (and consequently a different port number) so that it can continue to listen through the original server socket for connection requests while tending to the needs of the connected client. This scheme is helpful when two or more clients try to connect to a server simultaneously (a very common scenario).

Q: Why we write java.awt.event while importing Java.awt.*?

Answer: Package java.awt.event contains different event Listener Interfaces.

In java, events are represented by Objects these objects tell us about event and its source. Examples are: • ActionEvent (Clicking a button) • WindowEvent (Doing something with window e.g. closing, minimizing)

Q.How the problem of implementing all methods of interface was solved?

Answer: A class implementing an interface has to provide definitions of all methods present in that

Interface. To avoid giving implementations of all methods of an interface when we are not using these methods we use Event Adapter classes.

Main Point

Byte code • Java programs (Source code) are compiled into a form called Java byte codes.

- The Java compiler reads Java language source (.java) files, translates the source into Java bytecodes, and places the bytecodes into class (.class) files.
- The compiler generates one class file for each class contained in java source file.

Java Virtual Machine (JVM)• The central part of java platform is java virtual machine.

- Java bytecode executes by special software known as a "virtual machine".
- Most programming languages compile source code directly into machine code, suitable for execution

Java Program Development and Execution Steps Java program normally go through five phases. These are • Edit, • Compile, • Load, • Verify and • Execute

String Concatenation • "+" operator is used to concatenate strings

o System.out.println("Hello" + "World") will print Hello World on console

Comparing Strings: For comparing Strings never use == operator, use equals method of

String class. • == operator compares addresses (shallow comparison) while .equals compares values (deep comparison) o E.g. string1.equals(string2)

Primitives vs. Objects Everything in Java is an "Object", as every class by default inherits from class "Object", except a few primitive data types, which are there for efficiency reasons. o Primitive Data types of java ☐ boolean, byte ☐ 1 byte ☐ char, short ☐ 2 bytes

☐ int, float ☐ 4 bytes ☐ long, double ☐ 8 bytes

Primitive Data Type	Corresponding Object Class
byte	Byte
short	Short
int	Integer
long	Long
float	Float
double	Double
char	Character
boolean	Boolean

Stack vs. Heap: Stack and heap are two important memory areas. Primitives are created on the stack while objects are created on heap.

Wrapper Classes: Each primitive data type has a corresponding object (wrapper class). These wrapper classes provides additional functionality (conversion, size checking etc.), which a primitive data type cannot provide.

Converting Strings to Numeric Primitive Data Types

To convert a string containing digits to a primitive data type, wrapper classes can help. *parseXxx* method can be used to convert a String to the corresponding primitive data type.

- String value = "532"; int d = Integer.parseInt(value);

- String value = "3.14e6"; double d = Double.parseDouble(value);

The following table summarizes the parser methods available to a java programmer.

Data Type	Convert String using either ...
byte	Byte.parseByte(string) new Byte(string).byteValue()
short	Short.parseShort(string) new Short(string).shortValue()
int	Integer.parseInt(string) new Integer(string).intValue()
long	Long.parseLong(string) new Long(string).longValue()
float	Float.parseFloat(string) new Float(string).floatValue()
double	Double.parseDouble(string) new Double(string).doubleValue()

Selection & Control Structure:The if-else and switch selection structures are exactly similar to we have in C++. All relational operators that we use in C++ to perform comparisons are also available in java with same behavior. Likewise for, while and do-while control structures are alike to C++.

OOP Vocabulary Review

1 Classes:

- Definition or a blueprint of a user defined data type
- Prototypes for objects
- Think of it as a map of the building on a paper

2 Objects • Nouns, things in the world • Anything we can put a thumb on • Objects are instantiated or created from class

3 Constructor • A special method that is implicitly invoked. Used to create an Object (that is, an Instance of the Class) and to initialize it.

4 Attributes • Properties an object has

5 Methods • Actions that an object can do

Getters / Setters: The attributes of a class are generally taken as private or protected. So to access them outside of a class, a convention is followed known as getters & setters. These are generally public methods. The words *set* and *get* are used prior to the name of an attribute.

Static:A class can have static variables and methods. Static variables and methods are associated with the class itself and are not tied to any particular object. Therefore statics can be

accessed without instantiating an object. Static methods and variables are generally accessed by class name.

Garbage Collection & Finalize: Java performs garbage collection and eliminates the need to free objects explicitly. When an object has no references to it anywhere except in other objects that are also unreferenced, its space can be reclaimed. Before an object is destroyed, it might be necessary for the object to perform some action. For example: to close an opened file. In such a case, define a *finalize()* method with the actions to be performed before the object is destroyed.

1 Finalize: When a finalize method is defined in a class, Java run time calls *finalize ()* whenever it is about to recycle an object of that class.

Inheritance: In general, inheritance is used to implement a "is-a" relationship. Inheritance saves code rewriting for a client thus promotes reusability. In java parent or base class is referred as *super* class while child or derived class is known as *sub* class.

Polymorphism: "Polymorphic" literally means "of multiple shapes" and in the context of OOP, Polymorphic means "having multiple behaviors". • A parent class reference can point to the subclass objects because of is-a relationship. For example a Employee reference can point to:

o Employee Object o Teacher Object o Manager Object • A polymorphic method results in different actions depending on the object being Referenced o Also known as late binding or run-time binding.

Type Casting: In computer science, **type conversion** or **typecasting** refers to changing an entity of one data type into another. Type casting can be categorized into two types

1 Up-casting: • converting a smaller data type into bigger one

• Implicit - we don't have to do something special • No loss of information • Examples of
o Primitives

int a = 10; double b = a; o Classes Employee e = new Teacher();

2 Down-casting • converting a bigger data type into smaller one

• Explicit - need to mention • Possible loss of information • Examples of
o Primitives double a = 7.65; int b = **(int)** a;

o Classes Employee e = new Teacher();// up-casting

Teacher t= **(Teacher)** e;// down-casting

Types of Errors: Generally, you can come across three types of errors while developing software. These are Syntax, Logic & Runtime errors.

1 Syntax Errors: • Arise because the rules of the language are not followed.

2 Logic Errors: • Indicates that logic used for coding doesn't produce expected output.

3 Runtime Errors: • Occur because the program tries to perform an operation that is impossible to complete. • Cause exceptions and may be handled at runtime (while you are running the program) • For example divide by zero

What is an Exception?

• An exception is an event that usually signals an erroneous situation at run time • Exceptions are wrapped up as objects • A program can deal with an exception in one of three ways:

o ignore it o handle it where it occurs o handle it an another place in the program.

Types of Exceptions: Exceptions can be broadly categorized into two types, *Unchecked & Checked Exceptions*.

Unchecked Exceptions • Subclasses of Runtime Exception and Error. • Does not require explicit handling

• Run-time errors are internal to your program, so you can get rid of them by debugging your code

• For example, null pointer exception; index out of bounds exception; division by zero exception.

Checked Exceptions: • Must be caught or declared in a throws clause

• Compile will issue an error if not handled appropriately

• Subclasses of Exception other than subclasses of Runtime Exception.

• Other arrives from external factors, and cannot be solved by debugging

• Communication from an external resource - e.g. a file server or database

How Java handles Exceptions:Java handles exceptions via 5 keywords. **Try, catch, finally, throw & throws.**

1 try block:• Write code inside this block which could generate errors

2 Catch block: • Code inside this block is used for exception handling

• When the exception is raised from try block, only than catch block would execute.

3 finally block• This block always executes whether exception occurs or not.

4. throw• To manually throw an exception, keyword throw is used. **Note:** we are not covering throw clause in this handout

5 throws • If method is not interested in handling the exception than it can throw back the exception to the caller method using throws keyword. • Any exception that is thrown out of a method must be specified as such by a **throws** clause.

Abstract Classes: Abstract classes are used to define only part of an implementation. Because, information is not complete therefore an abstract class cannot be instantiate. However, like regular classes, they can also contain instance variables and methods that are fully implemented. The class that inherits from abstract class is responsible to provide details.

Interfaces: the implementation for these prototypes. All the methods inside an interface are abstract by default thus an interface is tantamount to a pure abstract class - a class with zero implementation. Interface can also contains static final constants

GUI classes vs. Non-GUI Support Classes: The classes present in the awt and swing packages can be classified into two broad categories. GUI classes & Non-GUI Support classes. The GUI classes as the name indicates are visible and user can interact with them. Examples of these are JButton, JFrame & JRadioButton etc The Non-GUI support classes provide services and perform necessary functions for GUI classes. They do not produce any visual output. Examples of these classes are Layout managers (discussed latter) & Event handling (see handout on it) classes etc.

Layout Managers: Layout Managers are used to form the appearance of your GUI. They are concerned with the arrangement of components of GUI Commonly used layout managers are

• Flow Layout • Grid Layout • Border Layout • Box Layout • Card Layout • GridBag Layout.

Inner Classes: • A class defined inside another class • Inner class can access the instance variables and members of outer class • It can have constructors, instance variables and methods, just like a regular class.

Anonymous Inner Classes

• has no name • same as inner class in capabilities • much shorter • Difficult to understand

Named vs. Anonymous Objects

Named • String s = "hello"; System.out.println(s); • "hello" has a named reference s.

Anonymous • System.out.println("hello");

Painting: Window is like a painter's canvas. All window paints on the same surface. More importantly, windows don't remember what is under them. There is a need to repaint when

portions are newly exposed.

18.1.2 Painting a Swing Component

Three methods are at the heart of painting a swing component like JPanel etc. For instance, paint() gets called when it's time to render -- then Swing further factors the paint()

call into three separate methods, which are invoked in the following order

protected void paintComponent(Graphics g)

protected void paintBorder(Graphics g)

protected void paintChildren(Graphics g)

An applet can react to major events in the following ways: • It can **initialize** itself. • It can **start** running. It can **stop** running.

- It can perform a **final cleanup**, in preparation for being unloaded

Socket Programming Definition: • A socket is one endpoint of a two-way communication link between two programs running generally on a network. • A socket is a bi-directional communication channel between hosts. A computer on a network often termed as host.

There are well-known **ports** which are

- o below 1024 o provides standard services
- o Some well-known ports are: ☐ FTP works on port 21 ☐ HTTP works on port 80 ☐ TELNET works on port 23 etc.

Close Socket: Don't forget to close the socket, when you finished.

Motivation • A lot of code involves boring conversion from a file to memory
o As you might recall that AddressBook program reads data from file and then parses it • This is a common problem.

Serializable Interface • By implementing this interface a class declares that it is willing to be read/written by automatic serialization machinery • Found in java.io package • Tagging interface - has no methods and serves only to identify the semantics of being serializable

Automatic Writing: • System knows how to recursively write out the state of an object to stream • If an object has the reference of another object, the java serialization mechanism takes

care of it and writes it too.

Automatic Reading: • System knows how to read the data from Stream and re-create object in memory • The recreated object is of type "Object" therefore Down-casting is required to convert it into actual type.

Object Serialization & Network: • You can read / write to a network using sockets.

- All you need to do is attach your stream with socket rather than file. • The class version should be same on both sides (client & network) of the network .

Preventing Serialization: • Often there is no need to serialize sockets, streams & DB connections etc because they do not represent the state of object, rather connections to external resources • To do so, **transient** keyword is used to mark a field that should not be serialized

- So we can mark them as, o **transient** Socket s; o **transient** OutputStream os; o **transient** Connection con; • Transient fields are returned as null on reading

Applets d included in a HTML page. ating system on which it runs ☐ An applet is a Panel that allows interaction with a Java program ☐ For security reasons, applets run in a

sandbox: they have no access to the client's file system **Applets Support** ☐ Most modern browsers support Java 1.4 if they have the appropriate plugin ☐ Sun provides an application appletviewer to view applets without using browser. ☐ In general you should try to write applets that can be run with any browser **What an App** ☐ You w

extending the class Applet or JApplet ☐ Applet is just a class like any other; you can even use it in applications if you want ☐ When ou write an applet, you are only writing part of a program ☐ The browser supplies the main method **The genealogy of applet:** The following figure shows the inheritance hierarchy of the JApplet class. This hierarchy determines much of what an applet can do and how, as you'll see on the next few pages.

bject | +----java.awt.Component | +----ava.awt.Container|+----java.awt.Panel | +----
java.applet.Applet|+----javax.swing.JApplet ☐ A small program written in Java an☐ It
is independent of the oper

How Client – Server Communicate :Normally, a server runs on a specific computer and has a socket that is bound to a specific port number. The server just waits, listening to the socket for a client to make a connection request. ☐ On the client side: The client knows the hostname of the machine on which the server is running and the port number to which the server is connected. ☐ On the server side, if the connection is accepted, a socket is successfully created and the client can use the socket to communicate with the server. ☐ Note that the socket on the client side is not bound to the port number used to make contact with the server. Rather, the client is assigned a port number local to the machine on which the client is running. ☐ The client and server can now communicate by writing to or reading from their sockets.

Steps – To Make a Simple Client To make a client, process can be split into 5 steps. These are:

1. **Import required package** You have to import two packages . java.net.*; . java.io.*;
2. **Connect / Open a Socket with:** Create a client socket (communication socket) Socket s = new Socket(“serverName”, serverPort) ;
3. **Get I/O Streams of Socket** Get input & output streams connected to your socket .
4. **Send / Receive Message** Once you have the streams, sending or receiving messages isn't a big task. It's very much similar to the way you did with files . To send messages pw.println(“hello world”); . To read messages ne();
5. **Close Socket** Don't forget to close the socket, when you finished your work s.c String recMsg = br.readLi lose();

Serialization in Java ☐ Java provides an extensive support for serialization ☐ Object knows how to read or write themselves to streams : ☐☐ You need to save and restore that state. ☐ The good news is that java serialization takes care of it automatically

Serializable Interface ☐ By implementing this interface a class declares that it is willing to be read/written by automatic serialization machinery ☐ Found in java.io.package ☐ Tagging interface – has no methods and serves only to identify the semantics of being serializable.

Automatic writing: System knows how to recursively write out the state of an object to stream ☐ If an object has the reference of another object, the java serialization mechanism takes care of it.

Automatic reading: System knows how to read the data from Stream and re-create object in memory. The recreated object is of type “Object” therefore Down-casting is required to convert it into actual type.