Q.2 a. What is the bytecode? How does Java solve both security and portability problems using bytecode?

Answer: Page Number of 9 Text Book

b. What is the difference between the logical operators: && and &? Explain with an example

Answer: Page Number of 72 Text Book

c. What is the enhanced 'for' loop syntax for arrays? Explain.

Answer: Page Number of 96 Text Book

Q.3 a. What is garbage collection? How is garbage collection performed in Java?

Answer:

In computer science, **garbage collection** (**GC**) is a form of automatic memory management. The garbage collector, or just collector, attempts to reclaim garbage, or memory occupied by objects that are no longer in use by the program.

Garbage collection is often portrayed as the opposite of manual memory management, which requires the programmer to specify which objects to deallocate and return to the memory system. However, many systems use a combination of approaches, including other techniques such as stack allocation and region inference.

Resources other than memory, such as network sockets, database handles, user interaction windows, and file and device descriptors, are not typically handled by garbage collection. Methods used to manage such resources, particularly destructors, may suffice to manage memory as well, leaving no need for GC. Some GC systems allow such other resources to be associated with a region of memory that, when collected, causes the other resource to be reclaimed; this is called *finalization*. Finalization may introduce complications limiting its usability, such as intolerable latency between disuse and reclaim of especially limited resources, or a lack of control over which thread performs the work of reclaiming.

Garbage collection in java :

1) **objects are created on heap in Java** irrespective of there scope e.g. local or member variable. whileits worth noting that class variables or static members are created in method area of Java memory space and both heap and method area is shared between different thread.
2) Garbage collection is a mechanism provided by Java Virtual Machine to **reclaim heap space** from objects which are **eligible for Garbage collection**.
3) **Garbage collection** relieves java programmer from **memory management** which is essential part of C++ programming and gives more time to focus on business logic.

4) **Garbage Collection in Java** is carried by a daemon thread called **Garbage Collector**.
5) Before removing an object from memory **Garbage collection thread invokes finalize () method** of that object and gives an opportunity to perform any sort of cleanup required.
6) You as Java programmer **can not force Garbage collection in Java**; it will only **trigger** if JVM thinks it needs a garbage collection **based on Java heap size**.
7) There are methods like **System.gc ()** and **Runtime.gc** () which **is used to send request of Garbage collection to JVM** but it's not guaranteed that garbage collection will happen.
8) If there is no memory space for creating new object in Heap **Java Virtual Machine** throws **OutOfMemoryError** or **java.lang.OutOfMemoryError heap space**
9) J2SE 5(Java 2 Standard Edition) adds a new feature called **Ergonomics** goal of ergonomics is to provide good performance from the **JVM** with minimum of command line **tuning**.

b. Differentiate between call by value and call by reference with the help of suitable example.

Answer:

Pass by value in java means passing a copy of the value to be passed. Pass by reference in java means the passing the address itself. In Java the arguments are always passed by value. Java only supports pass by value.

With Java objects, the object reference itself is passed by value and so both the original reference and parameter copy both refer to the same Java object. Java primitives too are passed by value.

```
public class Book {
String title;
public Book(String title){
this.title = title;
}
public void setTitle(String newTitle) {
this.title = newTitle;
}
public String getTitle() {
return this.title;
}
}
public class Example {
public static void main(String[] args){
Example example = new Example();
ArrayListbookList = new ArrayList();
bookList.add(0, new Book("First book"));
bookList.add(1, new Book("Second book"));
bookList.add(2, new Book("Third book"));
//we pass an object containing refrences :)
example.demonstrate(bookList);
//proof that its not only pass by value
System.out.println(bookList.get(0).getTitle());
}
public void demonstrate(ArrayListaBookList){
aBookList.get(0).setTitle("Changed the First title by refrence");
}
}
```

c. How is *'this'* used with constructors? Explain with example.

Answer:

* A Program to demonstrate the use of This keyword

```
 class ThisDemo
 {
    public ThisDemo()
    {
       this(10);
       System.out.println("First Constructor");
    }
    public ThisDemo(int a) // overloaded constructor
    {
       this(10,20);
       System.out.println("Second Constructor");
    }

    public ThisDemo( int a, int B) // another overloaded constructor
    {
       this("Prasad");
       System.out.println("Third Constructor");
    }
    public ThisDemo(String s) // and still another
    {
       System.out.println("Fourth Constructor");
    }



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

```
    ThisDemo first = new ThisDemo(); // one object should be created so that we can
 access other constructos through the constructor which is called first
    }
 }
```

Now this is the output of the program

OUTPUT

```
Fourth Constructor
Third Constructor
Second Constructor
First Constructor
```

Q.4 a. What is the difference between error and exception? What is user defined exception? Explain with example.

Answer:
Errors and exceptions both inherit from Throwable, but they differ in these ways:

Exceptions:

Can be checked or unchecked
Indicate an error caused by the programmer
Should be handled at the application level

Errors:

Are always unchecked
Usually indicate a system error or a problem with a low-level resource
Should be handled at the system level, if possible
User defined exception
Though Java provides an extensive set of in-built exceptions, there are cases in which we may need to define our own exceptions in order to handle the various application specific errors that we might encounter.
While defining an user defined exception, we need to take care of the following aspects:

- The user defined exception class should extend from Exception class.
- The toString() method should be overridden in the user defined exception class in order to display meaningful information about the exception.

NegativeAgeException.java

```
public class NegativeAgeException extends Exception {

privateint age;

publicNegativeAgeException(int age){
this.age = age;
   }
public String toString(){
return "Age cannot be negative" + " " +age ;
   }
}
```

b. What is the difference between Thread.start() and Thread.run()? Explain with example.

Answer:
Thread.start() method (native method) of Thread class actually does the job of running the Thread.run() method in a thread. If we directly call Thread.run() method it will executed in same thread, so does not solve the purpose of creating a new thread.

Q.5 a. What is an Applet? Differentiate between Applets and an application program

Answer:

A Java applet is an applet delivered to users in the form of Java bytecode. Java applets can run in a Web browser using a Java Virtual Machine (JVM), or in Sun's AppletViewer, a stand-alone tool for testing applets. Java applets are written in programming languages that compile to Java bytecode, usually in Java.

Since Java's bytecode is cross-platform or platform independent, Java applets can be executed by browsers for many platforms, including Microsoft Windows, Unix, Mac OS andLinux. It is also trivial to run a Java applet as an application with very little extra code. This has the advantage of running a Java applet in offline mode without the need for any Internet browser software and also directly from the integrated development environment (IDE).

The differences between an applet and an application are as follows:

1. Applets can be embedded in HTML pages and downloaded over the Internet whereas Applications have no special support in HTML for embedding or downloading.

2. Applets can only be executed inside a java compatible container, such as a browseror appletviewer whereas Applications are executed at command line by java.exe or jview.exe.

3. Applets execute under strict security limitations that disallow certain operations(sandbox model security) whereas Applications have no inherent security restrictions.

4. Applets don’t have the main() method as in applications. Instead they operate on anentirely different mechanism where they are initialized by init(),started by start(),stoppedby stop() or destroyed by destroy().

b. What is compareTo() function in Java? Explain with the help of a suitable example.

Answer: Page Number of 369 Text Book.

c. Write a program in Java to copy one file to another file using command line arguments.

Answer:

```
 import java.io.*;
public class CopyFile{
        private static void copyfile(String srFile, String dtFile){
                try{
                        File f1 = new File(srFile);
                        File f2 = new File(dtFile);
                        InputStream in = new FileInputStream(f1);
                        //For Append the file.
//                      OutputStream out = new FileOutputStream(f2,true);
                        //For Overwrite the file.
                        OutputStream out = new FileOutputStream(f2);
                        byte[] buf = new byte[1024];
                        intlen;
                        while ((len = in.read(buf)) > 0){
                                out.write(buf, 0, len);
                        }
                        in.close();
                        out.close();
                        System.out.println("File copied.");
                }
                catch(FileNotFoundException ex){
                        System.out.println(ex.getMessage() + " in the specified
directory.");
                        System.exit(0);
                }
                catch(IOException e){
                        System.out.println(e.getMessage());
                }
        }
```

```
    public static void main(String[] args){
        switch(args.length){
            case 0: System.out.println("File has not mentioned.");
                    System.exit(0);
            case 1: System.out.println("Destination file has not mentioned.");
                    System.exit(0);
            case 2: copyfile(args[0],args[1]);
                    System.exit(0);
            default :System.out.println("Multiple files are not allow.");
                    System.exit(0);
        }
    }
}
```

Q.6 a. Write a program in java to convert Array List to an array.

Answer

```
/*

Java ArrayList to String Array Example
        This Java ArrayList to String Array example shows how to convert ArrayList to String array
    in Java.
 */

import java.util.ArrayList;
import java.util.Arrays;

public class ArrayListToStringArrayExample {

    public static void main(String args[]){

        //ArrayList containing string objects
        ArrayList<String> aListDays = new ArrayList<String>();
        aListDays.add("Sunday");
        aListDays.add("Monday");
        aListDays.add("Tuesday");

        /*
         * To convert ArrayList containing String elements to String array, use
         * Object[] toArray() method of ArrayList class.
         *
           * Please note that toArray method returns Object array, not String array.
```

```
                 */

                //First Step: convert ArrayList to an Object array.
                Object[] objDays = aListDays.toArray();

                //Second Step: convert Object array to String array
                      String[] strDays = Arrays.copyOf(objDays, objDays.length, String[].class);

                System.out.println("ArrayList converted to String array");

                //print elements of String array
                for(int i=0; i < strDays.length; i++){
                    System.out.println(strDays[i]);
                }
        }
}

/*
Output of above given ArrayList to String Array example would be
ArrayList converted to String array
Sunday
Monday
Tuesday
*/
```

b. What are the differences between Swing and AWT?

Answer:
AWT is a heavy weight component but swing is light weight component. AWT is OS dependent but Swing is not OS dependent.
Swing is not a complete replacement for the AWT-it is built on top of the AWT architecture. Swing simply gives you more capable user interface components.

Even though AWT is powerful, there are many compelling reasons as to why Swings is a more successful and preferred choice for UI programmers. They are:

1. Swing has a rich and convenient set of user interface elements.
2. Swing has few dependencies on the underlying platform. This means that it is less prone to platform-specific bugs.
3. Swing gives a consistent user experience across platforms

Q7 a. What are the major tasks in the overall website development process? What knowledge and skills do you expect to learn to participate in this process?

Answer: Page Number 22 of Text Book

b. Describe the main differences between HTML and XHTML.

Answer:

HTML is rapidly being replaced by XHTML. The differences are very minor, but the results of switching can be worth the effort. The primary benefit is that XHTML is more widely accepted in non "computer" devices like cell phone, palm devices and other scaled down browsers. This is commonly called portability between devices. XHTML is also said to be extensible, which is the fancy way of saying the new tags can be added without a new document type declaration.

Differences:

- the tags in the page **MUST** be in lower case, so instead of <IMG SRC="resource/frankisboat.gif" WIDTH="389" HEIGHT="227" BORDER="0" ALT="boat">, as we would do in HTML, we instead use: <imgsrc="resource/frankisboat.gif" width="389" height="227" border="0" alt="boat" />
- all tags must close, either by using a corresponding closing tag which has a slash, like paragraph (<p></p>) for example, or some tags are self closing like the above imgsrc tag and line break (<br />). In HTML, many of these tags were simply left open.
- all tags must be properly nested, so if you start tag "a" and then start tag "b", you must close tag "b" before you close tag "a"
- some tags which were previously allowed are no longer allowed, although see the discussion below of document type declarations (DTD's).
- all attributes must also be lowercase
- all values for attributes must be encased in single or double quotes
- all attributes must be long form, not abbreviated, for example: disabled="true" in XHTML vs DISABLED in HTML
- structure must be separated from content. So for example, the <p> tag is a content tag (paragraph) so you can't put a table in it for example, because a table is a format construct. You can however put the <p> tag inside <td></td> tags with no problem because the content goes in the construct, not the other way around.

The first thing you will notice if you look at the source of an XHTML document is that the first line is a document type declaration (DTD also called the **DOCTYPE**). There are three that are used, strict (that will only validate if you have no deprecated tags), transitional (which will still validate with deprecated tags) and frameset (which is for a page that "sets" up "frames"). Oddly enough, even though all tags in XHTML are lower case, parts of the DTD must be in upper case. The three DTD's look like this:

- Strict -
  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "DTD/xhtml1-strict.dtd">
- Transitional -
  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
- Frameset -
  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN" "DTD/xhtml1-frameset.dtd">

b)If you have just finished a nex XTML document and you want to make sure that confirms with the XHTML specification you can use the XHTML online validator at http://validator.w3.org/.
There are three ways you can use this validator:

- By URL – Specific the URL where your XHTML document I slocated and click “check”
- By file upload – Use the “Browse” to locate an XHTML document on your local hard disk and click “Check”.
- By direct input – Copy and paste the content of an XTML document to the input area and click “Check”.

The validator will return a page with validation result.

c. What role does white space play in an HTML file? What constitutes white space in an HTML document?

Answer: Page Number of 41 Text Book

Q8. a. Define design. What are the elements of a good design?

Answer:
The elements of design are the building blocks of design. These elements are what make up every page you build. And understanding the basic elements you'll be able to put together more powerful Web pages.

There are five basic elements of every design:

<u>Lines and Linework</u>

Lines include borders and rules. They can be horizontal or vertical and help delineate the spaces around elements on Web pages. Linework increases the readability of the design.

Shape

Shapes make up any enclosed contour in the design. Shapes on most Web pages are square or rectangular. But they don't have to be. You can use images to generate other shapes within your designs.

Texture

Texture gives a design a feeling of surface. Texture on Web pages is all visual, but you can use natural textures or artificial to get the effect in your designs.

Color

Color is the one design element that most Web designers are acutely aware of. But remember that color is not a required element of any design. In fact, a good plan in design is to create the design without color first, then add as little color as you can to enhance the design.

Direction

Direction gives your Web designs motion. In most designs there is a sense of movement in a direction across the design. Good designs lead the eye through the design in a deliberate fashion so that the viewer sees what the designer wants.

b. With the help of example discuss the different attributes available for table?

Answer: Page Number 77 of Text Book

c. List the six concrete steps needed to create the site's information architecture.

Answer: Page Number 140-141 of Text Book

Q9 a. what are HTTP GET and POST methods? Explain briefly.

Answer:

```
echo Content-type: text/html
echo ""

/bin/cat << EOM
<HTML>
<HEAD><TITLE>File Output: /home/user1/public_html/text-file.txt </TITLE>
</HEAD>
```

```
<BODY bgcolor="#cccccc" text="#000000">
<HR SIZE=5>
<H1>File Output: /home/user1/public_html/text-file.txt </H1>
<HR SIZE=5>
<P>
<SMALL>
<PRE>
EOM

/bin/cat /home/user1/public_html/text-file.txt

CAT << EOM
</PRE>
</SMALL>
<P>
</BODY>
</HTML>
```

HTML Source:

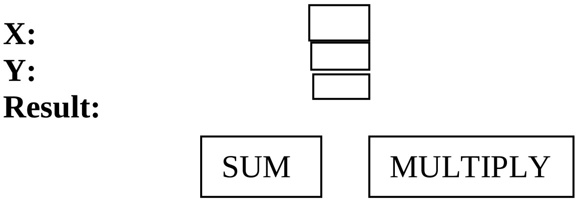
```
<HTML>
<HEAD><TITLE>Test ISINDEX HTML tag</TITLE></HEAD>
<BODY bgcolor="#cccccc" text="#000000">
<H2>Test ISINDEX HTML tag</H2>
<ISINDEX prompt="Enter value:" action="http://localhost/cgi-bin/catpage">
</BODY>
</HTML>
```

Text entered: /tmp/text-file.txt
The following will get generated: http://localhost/cgi-bin/catpage?%2Ftmp%2Ftext-file.txt
The CGI will then spit out the text page specified.

b. Explain the following text input elements in a form with the help of examples: (i) Radio Buttons (ii) Check Boxes

Answer: Page Number 281 & 282 of Text Book

c. Write a Java Script to do the following:



After entering the values of X and Y, if SUM or MULTIPLY button is clicked then sum or multiplication respectively of X and Y should be printed

Answer:

```
<html>
  <head>
      <title>Input tutorial</title>
      <script language="javascript">
              function addNumbers()
              {
                      var X = parseInt(document.getElementById("X").value);
                      var Y = parseInt(document.getElementById("Y").value);
                      var ansD = document.getElementById("result");
                      ansD.value = X + Y;
              }
function MultNumbers()
              {
                      var X = parseInt(document.getElementById("X").value);
                      var Y = parseInt(document.getElementById("Y").value);
                      var ansD = document.getElementById("result");
                      ansD.value = X*Y;
              }

      </script>
  </head>
  <body>
      value1 = <input type="text" id="X" name="X" value="1"/>
      value2 = <input type="text" id="Y" name="Y" value="2"/>
      <input type="button" name="Sumbit" value="Sum"
onclick="javascript:addNumbers()"/>
      Answer = <input type="text" id="Result" name="Result" value=""/>
<input type="button" name="Sumbit" value="Multiply"
onclick="javascript:MultNumbers()"/>

  </body>
</html>
```

Text Books

(1) **The Complete Reference Java, Herbert Schildt, TMH, 7th Edition, 2007.**

(2) **An Introduction to Web Design + Programming, Paul S.Wang and Sanda S. Katila, Thomson Course Technology, India Edition, 2008.**