String handling

1. Description
    a. String class is immutable
    b. String is a final class, no other class can extend it, and you cannot change the state of the string
    c. String values cannot be compare with '==', for string value comparison, use equals() method

2. String Initialization
    a. String abc = "This is a string object";
    b. String bcd = new String("this is also string object");
    c. char[] c = {'a','b','c','d'};
    d. String cdf = new String(c);
    e. String junk = abc+" This is another String object";

3. convert Character array to String object
    a. char ch[] = {'M','y',' ','J','a','v','a',' ','e','x','a','m','p','l','e'};
    b. String chStr = String.copyValueOf(ch);

    c. String subStr = String.copyValueOf(ch,3,4);
    d. System.out.println(subStr);

4. append or concat two Strings
    a. String b = "jump ";
    b. String c = "No jump";

    c. String d = b+c;

    d. d = b.concat(c);

5. compare two String objects

    a. String x = "JUNK";
    b. String y = "junk";

    c. x.equals(y)
    d. x.equalsIgnoreCase(y)

6. compare StringBuffer object to String object

    a. String c = "We are comparing the content with a StringBuffer content";
    b. StringBuffer sb = new StringBuffer("We are comparing the content with a StringBuffer content");
    c. c.contentEquals(sb)

7. get byte array from a string object
    a. String str = "core java api";
    b. byte[] b = str.getBytes();

8. get index of a character or string from another String

    a. String str = "Use this string for testing this";

b. System.out.println("Basic indexOf() example");
c. System.out.println("Char 's' at first occurance: "+str.indexOf('s'));
d. System.out.println("String \"this\" at first occurance: "+str.indexOf("this"));

e. Output:
i. Basic indexOf() example
ii. Char 's' at first occurance: 1
iii. String "this" at first occurance: 4

9. String lastIndexOf()
a. String str = "Use this string for testing this";
b. System.out.println("Basic lastIndexOf() example");
c. System.out.println("Char 's' at last occurance: "+str.lastIndexOf('s'));
d. System.out.println("String \"this\" at last occurance: "+str.lastIndexOf("this"));
e. System.out.println("first occurance of char 's' from 24th index backwards: "+str.lastIndexOf('s',24));
f. System.out.println("First occurance of String \"this\" from 26th index backwards: "+str.lastIndexOf("this",26));

10. find a string start with another string

a. String str = "This is an example string.";
b. System.out.println("Is this string starts with \"This\"? "+str.startsWith("This"));
c. System.out.println("Is this string starts with \"is\"? "+str.startsWith("is"));
d. System.out.println("Is this string starts with \"is\" after index 5? "+str.startsWith("is", 5));

11. find a string ends with another string
a. String str = "This is a java string example";
b. if(str.endsWith("example")){
c. System.out.println("This String ends with example");
d. } else {
e. System.out.println("This String is not ending with example");
f. }

12. break or split a string with a delimiter

a. String str = "This program splits a string based on space";
b. String[] tokens = str.split(" ");
c. tokens = str.split("\\s+");

13. extract Char Array from String
a. String str = "Copy chars from this string";
b. char[] ch = new char[5];
c. /**
i. The getChars() method accepts 4 parameters
ii. first one is the start index from string
iii. second one is the end index from string
iv. third one is the destination char array
v. forth one is the start index to append in the char array.
d. */
e. str.getChars(5, 10, ch, 0);
f. System.out.println(ch);

14. replace string characters

    a. String str = "This is an example string";
    b. System.out.println("Replace char 's' with 'o':"+str.replace('s', 'o'));
    c. System.out.println("Replace first occurance of string\"is\" with \"ui\":"+str.replaceFirst("is", "ui"));
    d. System.out.println("Replacing \"is\" every where with \"no\":"+str.replaceAll("is", "no"));

15. change case of a string characters
    a. String str = "Change My Case";
    b. System.out.println("Upper Case: "+str.toUpperCase());
    c. System.out.println("Lower Case: "+str.toLowerCase());

16. trim spaces in the given string

    a. String str = " Junkof garbage ";
    b. System.out.println(str.trim());

    c. Output:
        i. “Junkof garbage”

17. format given string
    a. String str = "This is %s format example";
    b. System.out.println(String.format(str, "string"));
    c. String str1 = "We are displaying no %d";
    d. System.out.println(String.format(str1, 10));

    e. /**
        - String format by specifying Locale details
    f. */
    g. System.out.println("String format with Locale info:");
    h. System.out.println(String.format(Locale.US, str1, 10));

18. match a format in string using regular expression
    a. String str = “www.java2novice.com”;
    b. str[i].matches("^www\\.(.+)");

    c. True

19. remove multiple spaces in a string
    a. import java.util.StringTokenizer;

    b. String str = "String With Multiple Spaces";
    c. StringTokenizer st = new StringTokenizer(str, " ");
    d. StringBuffer sb = new StringBuffer();
    e. while(st.hasMoreElements()){
    f. sb.append(st.nextElement()).append(" ");
    g. }
    h. System.out.println(sb.toString().trim());

20. remove non-ascii characters from a string
    a. String str = "Bj��rk����oacute�";
    b. str = str.replaceAll("[^\\p{ASCII}]", "");

21. remove html tags from a string
    a. String text = "<B>I don't want this to be bold<\\B>";
    b. text = text.replaceAll("\\<.*?\\>", "");

22. get **line count** from a string
    a. public static int getLineCount(String text){
        i. return text.split("[\n|\r]").length;
    b. }

    c. String str = "line1\nline2\nline3\rline4";
    d. int count = getLineCount(str);

    e. Output:
        i. 4

23. ArrayList to String
    a. ArrayList<String> aListDays = new ArrayList<String>();
    b. aListDays.add("Sunday");
    c. aListDays.add("Monday");
    d. aListDays.add("Tuesday");

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

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

24. String Reverse
    a. String strOriginal = "Hello World";
    b. String strReversed= new StringBuffer(strOriginal).reverse().toString();

25. String to Date
    a. import java.text.ParseException;
    b. import java.text.SimpleDateFormat;
    c. import java.util.Date;

    d. String strDate = "21/08/2011";
    e. SimpleDateFormat sdf = new SimpleDateFormat("dd/MM/yyyy");
    f. Date date = sdf.parse(strDate);

26. String to ArrayList
    a. String strNumbers = "1,2,3,4,5";
    b. String[] strValues = strNumbers.split(",");
    c. /*

- Use asList method of Arrays class to convert Java String array to ArrayList

d. */

e. ArrayList<String> aListNumbers = new ArrayList<String>(Arrays.asList(strValues));

27. Substring Example

a. /*

b. This will print the substring starting from index 6

c. */

d. System.out.println(name.substring(6));

e. /*

f. This will print the substring starting from index 0 upto 4 not 5.

g. */

h. System.out.println(name.substring(0,5));