Core Java Assignment 3

Assignment on String Class

1.Write an application to determine the length of the string =”Hello world”.

**package** PracticeString;

**public** **class** LengthStr {

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

{

String str="Hello wrold";

System.***out***.println(str.length());

}

}

Output:

Length of String : 11

2.Write an application to join the two string “Hello” and “How are you?”

**package** PracticeString;

**public** **class** JoinStr {

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

{

String s1="Hello";

String s2="How are you?";

String s=String.*join*(" ",s1,s2);

System.***out***.println(s);

}

}

Output:

Hello How are you?

3. Given a String "Java String pool refers to collection of Strings which are stored in heap memory", perform the following operations (Hint: all operation can be performed using String methods)

a. Print the string to console in lowercase

b. Print the string to console in uppercase

c. Replace all'a' character in the string with S sign

d. Check if the original String contains the word "collection"

e. Check if the following String "java string pool refers to collection of strings which are stored in heap memory" matches the original

f. If the string does not match check if there is another method which can be used to check if the strings are equal

**package** PracticeString;

**public** **class** StrOperations {

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

{

String str="Java String Pool Refers To Collection Of String Which Are Store In Heap Memory";

String nstr="java String pool refers to collection of String which are store in heap memory";

String cont="Collection";

String str1=str.toLowerCase();

String str2=str.toUpperCase();

String reStr=str.replace("a", "$");

**boolean** ans= str.contains(cont);

System.***out***.println("Original String:" +" "+str);

System.***out***.println("Lower case String:" +" "+str1);

System.***out***.println("Upper case String:" +" "+str2);

System.***out***.println("Replaced : " +" "+reStr);

System.***out***.println("Check : " +" "+ans);

System.***out***.println(str.matches(nstr));

System.***out***.println(str.compareTo(nstr));

System.***out***.println(str.compareToIgnoreCase(nstr));

System.***out***.println(str.equals(nstr));

System.***out***.println(str.equalsIgnoreCase(nstr));

}

}

Output:

Original String: Java String Pool Refers To Collection Of String Which Are Store In Heap Memory

Lower case String: java string pool refers to collection of string which are store in heap memory

Upper case String: JAVA STRING POOL REFERS TO COLLECTION OF STRING WHICH ARE STORE IN HEAP MEMORY

Replaced : J$v$ String Pool Refers To Collection Of String Which Are Store In He$p Memory

Check : true

false

-32

0

false

true

Assignment on String Buffer Class

STRINGBUFFER

1. Write an application to append the following strings “StringBuffer”, “is a peer of a String”, “that provides much of”, “the functionalities of string” using StringBuffer.

**package** PracticeString;

**public** **class** Apppend {

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

{

StringBuffer sb=**new** StringBuffer("StringBuffer is a peer class of a string");

sb.append(" " + "that provides much of");

sb.append(" " +" the fuctionalities of string");

System.***out***.println(sb);

}

}

}

Output:

StringBuffer is a peer class of a string that provides much of the fuctionalities of string

1. Insert the following string “insert text” into the string “it is used to \_at the specified index position” at the location denoted by \_sign using StringBuffer.

**package** PracticeString;

**public** **class** InsertText {

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

{

StringBuffer sb=**new** StringBuffer("It is used to at the specified index position");

sb.insert(13, "insert text ");

System.***out***.println(sb);

}

}

Output:

It is used to insert text at the specified index position

1. Reverse the following string “This method is used to return the reverse object on which it was called” using StringBuffer class

**package** PracticeString;

**public** **class** StrReverse {

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

{

StringBuffer sb=**new** StringBuffer("This method returns the reverse object on which it was called");

sb.reverse();

System.***out***.println(sb);

}

}

Output:

dellac saw ti hcihw no tcejbo esrever eht nruter ot desu si dohtem sihT

1. Write an application to append the following strings “StringBuilder”, “is a peer of a String”, “that provides much of”, “the functionalities of string” using StringBuilder.

**package** PracticeString;

**public** **class** Apppend1 {

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

{

StringBuilder sb=**new** StringBuilder("StringBuffer is a peer class of a string");

sb.append(" " + "that provides much of");

sb.append(" " +" the fuctionalities of string");

System.***out***.println(sb);

}

}

}

Output:

StringBuilder is a peer class of a string that provides much of the fuctionalities of string

1. Insert the following string “insert text” into the string “it is used to \_at the specified index position” at the location denoted by \_sign using StringBuilder.

**package** PracticeString;

**public** **class** InsertText1 {

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

{

StringBuilder sb=**new** StringBuilder ("It is used to at the specified index position");

sb.insert(13, "insert text ");

System.***out***.println(sb);

}

}

Output:

It is used to insert text at the specified index position

1. Reverse the following string “This method is used to return the reverse object on which it was called” using StringBuilder class

**package** PracticeString;

**public** **class** StrReverse {

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

{

StringBuilder sb=**new** S StringBuilder ("This method returns the reverse object on which it was called");

sb.reverse();

System.***out***.println(sb);

}

}

Output:

dellac saw ti hcihw no tcejbo esrever eht nruter ot desu si dohtem sihT