**LAB # 01**

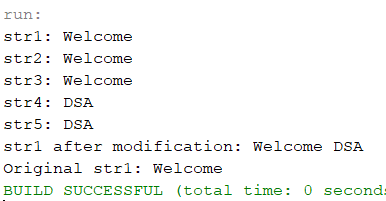
**INTRODUCTION TO STRING POOL, LITERALS, AND WRAPPER CLASSES**

## LAB TASKS

1. Write a program that initialize five different strings using all the above mentioned ways, i.e., a)string literals b)new keyword . also use intern method and show string immutability.

**Code: Output:**

public class Task1{

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

String str1 = "Welcome";

System.out.println("str1: " + str1);

String str2 = new String("Welcome");

System.out.println("str2: " + str2);

String str3 = "Welcome".intern();

System.out.println("str3: " + str3);

String str4 = "DSA";

System.out.println("str4: " + str4);

String str5 = new String("DSA");

System.out.println("str5: " + str5);

str1 = str1 + " DSA";

System.out.println("str1 after modification: " + str1);

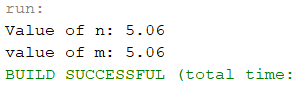
System.out.println("Original str1: " + "Welcome");  
 }

}

1. Write a program to convert primitive data type Double into its respective wrapper object.

**Code:**

public class Task2 { **Output:**

 public static void main(String[] args) {

double n = 5.06;

System.out.println("Value of n: " + n);

Double m = n;

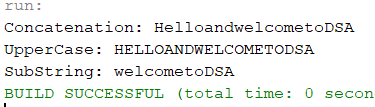
System.out.println("value of m: " + m);

}

}

3) Write a program that initialize five different strings and perform the following operations. a)Concatenate all five stings. b) Convert fourth string to uppercase. c) Find the substring from the concatenated string from 8 to onward

**Code: Output:**

public class Task3 {

public static void main(String[] args) {

String str1= "Hello";

String str2= "and";

String str3= "welcome";

String str4= "to";

String str5= "DSA";

String str6= str1+str2+str3+str4+str5;

System.out.println("Concatenation: "+ str6);

System.out.println("UpperCase: "+ str6.toUpperCase());

System.out.println("SubString: "+ str6.substring(8));

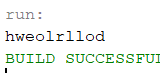
}

}

4.You are given two strings word1 and word2. Merge the strings by adding letters in alternating order, starting with word1. If a string is longer than the other, append the additional letters onto the end of the merged string. Return *the merged string.*

**Code:**

public class Task4 { **Output:**

 public static String mergeString(String w1, String w2) {

StringBuilder merged = new StringBuilder();

int i = 0, j = 0;

while (i < w1.length() && j < w2.length()) {

merged.append(w1.charAt(i));

merged.append(w2.charAt(j));

i++;

j++; }

while (i < w1.length()) {

merged.append(w1.charAt(i));

i++; }

while (j < w2.length()) {

merged.append(w2.charAt(j));

j++; }

return merged.toString(); }

public static void main(String[] args) {

String w1 = "hello";

String w2 = "world";

String result = mergeString(w1, w2);

System.out.println(result);

}

}

5. Write a Java program to find the minimum and maximum values of Integer**,** Float**,** and Double using the respective wrapper class constants.

**Code:**

public class Task5 {

public static void main(String[] args) {

System.out.println("Integer Minimum Value: " + Integer.MIN\_VALUE);

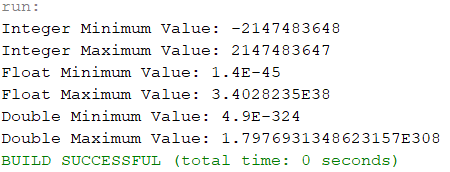
System.out.println("Integer Maximum Value: " + Integer.MAX\_VALUE);

System.out.println("Float Minimum Value: " + Float.MIN\_VALUE);

System.out.println("Float Maximum Value: " + Float.MAX\_VALUE);

System.out.println("Double Minimum Value: " + Double.MIN\_VALUE);

System.out.println("Double Maximum Value: " + Double.MAX\_VALUE);

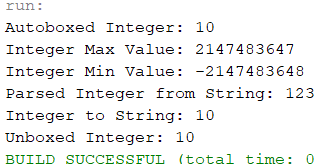
 }

} **Output:**

## HOME TASKS

1. Write a JAVA program to perform Autoboxing and also implement different methods of wrapper class.

**Code: Output:**

public class Homelab1 {

public static void main(String[] args) {

int primitiveInt = 10;

Integer wrapperInt = primitiveInt;

System.out.println("Autoboxed Integer: " + wrapperInt);

System.out.println("Integer Max Value: " + Integer.MAX\_VALUE);

System.out.println("Integer Min Value: " + Integer.MIN\_VALUE);

System.out.println("Parsed Integer from String: " + Integer.parseInt("123"));

System.out.println("Integer to String: " + Integer.toString(wrapperInt));

int unboxedInt = wrapperInt;

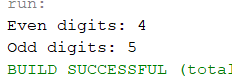
System.out.println("Unboxed Integer: " + unboxedInt);

}

}

1. Write a Java program to count the number of even and odd digits in a given integer using Autoboxing and Unboxing.

**Code: Output:**

import java.util.ArrayList;

import java.util.List;

public class Homelab1 {

public static void main(String[] args) {

int number = 123456789;

List<Integer> digits = new ArrayList<>();

while (number > 0) {

digits.add(number % 10);

number /= 10;

}

int evenCount = 0;

int oddCount = 0;

for (Integer digit : digits) {

int d = digit;

if (d % 2 == 0) {

evenCount++;

} else {

oddCount++;

} }

System.out.println("Even digits: " + evenCount);

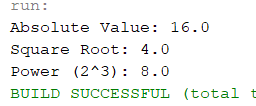
System.out.println("Odd digits: " + oddCount);

}

}

1. Write a Java program to find the absolute value, square root, and power of a number using Math classmethods, while utilizing Autoboxing and Wrapper classes**.**

**Code: Output**

public class Homelab1 {

public static void main(String[] args) {

Double number = -16.0;

Double absoluteValue = Math.abs(number);

Double squareRoot = Math.sqrt(Math.abs(number));

Double power = Math.pow(2, 3);

System.out.println("Absolute Value: " + absoluteValue);

System.out.println("Square Root: " + squareRoot);

System.out.println("Power (2^3): " + power);

}

}

1. Write a Java program to **reverse only the vowels** in a string.

**Code: Output:**

public class Homelab1 {

public static void main(String[] args) {

String input = "Hello World";

String reversedVowels = reverseVowels(input);

System.out.println("Reversed Vowels: " + reversedVowels);

}

public static String reverseVowels(String s) {

char[] chars = s.toCharArray();

int left = 0, right = s.length() - 1;

String vowels = "aeiouAEIOU";

while (left < right) {

while (left < right && vowels.indexOf(chars[left]) == -1) {

left++; }

while (left < right && vowels.indexOf(chars[right]) == -1) {

right--; }

if (left < right) {

char temp = chars[left];

chars[left] = chars[right];

chars[right] = temp;

left++;

right--;

} }

return new String(chars);

}

}

1. Write a Java program to **find the longest word** in a sentence.

**Code:**

public class Homelab1 {

public static void main(String[] args) {

String sentence = "Write a Java program to find the longest word in a sentence";

String longestWord = findLongestWord(sentence);

System.out.println("Longest Word: " + longestWord); } **Output:**

 public static String findLongestWord(String sentence) {

String[] words = sentence.split(" ");

String longest = "";

for (String word : words) {

if (word.length() > longest.length()) {

longest = word;

} }

return longest;

}

}