**LAB 6**

**Solve below Questions:**

**1.     Write the programme to sort the integers 8, 4, 3,5,6 and the alphabetical string C, O, I, P, U, in ascending order. Show the resulting output.**

**PROGRAM:**

**package** Lab6;

**import** java.util.Arrays;

**public** **class** SortedElements {

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

// Sorting integers

**int**[] intArr = {8, 4, 3, 5, 6};

Arrays.*sort*(intArr);

System.***out***.println("Sorted integers: " + Arrays.*toString*(intArr));

// Sorting strings

String[] strArr = {"C", "O", "I", "P", "U"};

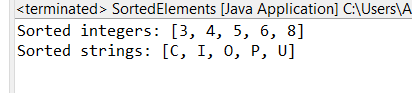
Arrays.*sort*(strArr);

System.***out***.println("Sorted strings: " + Arrays.*toString*(strArr));

}

}

**OUTPUT:**

****

**2.     Write a Java program to implement the bubble sort algorithm to sort an array of integers in ascending order.**

**PROGRAM:**

**package** mDemo;

**public** **class** BubbleSortArray {

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

// Array of integers

**int** nums[] = {5, 6 ,12, 45, 7, 87, 22, 1, 0 };

**int** size = nums.length; // Size variable to store the size of the array

**int** temp = 0; // Temporary Variable for Swapping of variable

System.***out***.println("Sorting stages: ");

// Loop to Iterate through array

**for**(**int** i = 0; i < size ; i++) {

**for**(**int** j = 0; j < (size - i) - 1; j++ ) { // Loop to Iterate through only range that requires swapping

// If statement to swap elements if initial element is greater than its next element

**if** (nums[j] > nums[j+1]) {

temp = nums[j];

nums[j] = nums[j+1];

nums[j+1] = temp;

}

}

// Printing array after every iteration

**for**(**int** num : nums) {

System.***out***.print(num + " ");

}

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

}

System.***out***.println("After Bubble Sort: ");

// Printing array after sorting

**for**(**int** num : nums) {

System.***out***.print(num + " ");

}

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

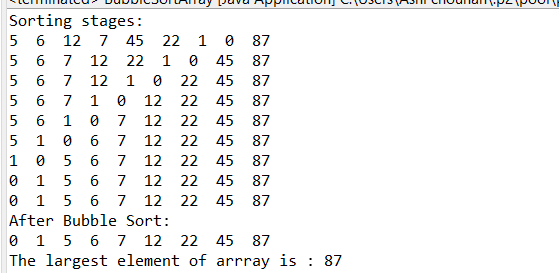
// Largest element of array

System.***out***.println("The largest element of arrray is : " + nums[size-1]);

}

}

**OUTPUT:**

****

**3.     Write a program to input an array 10 elements and print the cube of prime numbers in it.**

**PROGRAM:**

**package** Lab6;

**import** java.util.Scanner;

**public** **class** PrimeNumberCubeArray {

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

// Creating a Scanner object for user input

Scanner sc = **new** Scanner(System.***in***);

System.***out***.print("Enter the size of array: ");

**int** n = sc.nextInt();

// Initialize the array with size n

**int**[] nums = **new** **int**[n];

// Read elements into the array

**for** (**int** i = 0; i < n; i++) {

System.***out***.print("Enter element " + (i + 1) + ": ");

nums[i] = sc.nextInt();

}

System.***out***.println("Prime numbers in the array:");

// Check for prime numbers and print them

**for** (**int** i = 0; i < n; i++) {

**boolean** isPrime = **true**;

// Check if the number is prime

**if** (nums[i] <= 1) {

isPrime = **false**;

} **else** {

**for** (**int** j = 2; j <= Math.*sqrt*(nums[i]); j++) {

**if** (nums[i] % j == 0) {

isPrime = **false**;

**break**;

}

}

}

// Print the number if it is prime

**if** (isPrime) {

System.***out***.println(nums[i]);

}

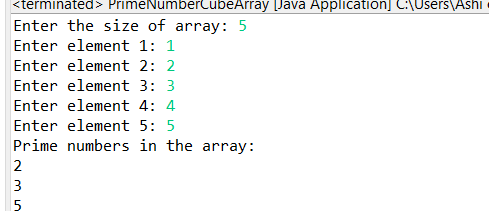
}

sc.close();

}

}

**OUTPUT:**

****

**4.     Write a java program to implement integer wrapper class methods.(any 3 methods)**

**PROGRAM:**

**package** Lab6;

**public** **class** WrapperClassMethods {

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

**int** a = 100;

// Method 1 - Integer.valueOf()

Integer num1 = Integer.*valueOf*(a); // Integer.valueOf() assigns value to a wrapper class object

System.***out***.println("Integer num1: " + num1);

// Method 2 - intValue()

**int** intPrimitive = num1.intValue(); // This function converts wrapper class object value to int value

System.***out***.println("Primitive int value obtained from Integer object: " + intPrimitive);

// Method 3 - parseInt(String)

String numStr = "73739338";

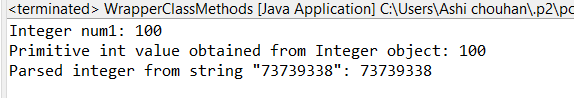
**int** parsedInt = Integer.*parseInt*(numStr); // Converts Number String value into integer

System.***out***.println("Parsed integer from string \"" + numStr + "\": " + parsedInt);

}

}

**OUTPUT:**

****

**5.     Write a java program to implement double wrapper class methods.(any 3 methods)**

**PROGRAM:**

**package** Lab6;

**public** **class** DoubleWrapperClassMethods {

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

// Method 1 - valueOf(double d)

Double doubleNum = Double.*valueOf*(25.78); // Assigns value to a double wrapper class object

System.***out***.println("Double object created with valueOf: " + doubleNum);

// Method 2 - doubleValue()

**double** doublePrimitive = doubleNum.doubleValue(); // Wrapper class object value into primitive value

System.***out***.println("Primitive double value obtained from Double object: " + doublePrimitive);

// Method 3 - parseDouble(String s)

String doubleString = "27483.264";

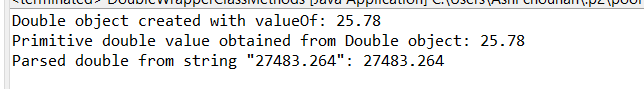
**double** parsedDouble = Double.*parseDouble*(doubleString); // Converts Double stirng value into a double value

System.***out***.println("Parsed double from string \"" + doubleString + "\": " + parsedDouble);

}

}

**OUTPUT:**

****

**6.     Write a java program to implement float wrapper class methods.(any 3 methods)**

**PROGRAM:**

**package** Lab6;

**public** **class** FloatWrapperClassMethods {

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

// Method 1 - valueOf(float f)

Float floatNum = Float.*valueOf*(10.75f); // Assigns value to a float wrapper class object

System.***out***.println("Float object created with valueOf: " + floatNum);

// Method 2 - floatValue()

**float** floatPrimitive = floatNum.floatValue(); // Wrapper class object value into primitive value

System.***out***.println("Primitive float value obtained from Float object: " + floatPrimitive);

// Method 3 - parseFloat(String s)

String floatString = "123.456";

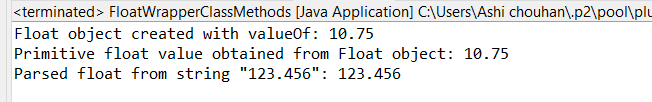
**float** parsedFloat = Float.*parseFloat*(floatString); // Converts float string value into a float value

System.***out***.println("Parsed float from string \"" + floatString + "\": " + parsedFloat);

}

}

**OUTPUT:**

****

**7.     Write a Java program to validate email addresses using regular expressions. The email should have the format username@domain.com where username and domain can contain alphanumeric characters, dots, and hyphens.**

**PROGRAM:**

**package** Lab6;

**import** java.util.Scanner;

**import** java.util.regex.Pattern;

**public** **class** RegexValidateEmail {

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

// Regex for email

String emailRegex = "^[a-zA-Z0-9.\_-]+@[a-zA-Z0-9.-]+\\.[a-zA-Z]{2,}$";

// Compiling the regex

Pattern p = Pattern.*compile*(emailRegex);

Scanner sc = **new** Scanner(System.***in***);

// Taking user input for email using scanner object

System.***out***.print("Enter an email address: ");

String email = sc.nextLine();

// Boolean variable to check if entered email matches the regex pattern

**boolean** isValidEmail = p.matcher(email).matches();

// Printing valid and invalid email based on boolean variable value

**if** (isValidEmail) {

System.***out***.println(email + " is a valid email address.");

} **else** {

System.***out***.println(email + " is not a valid email address.");

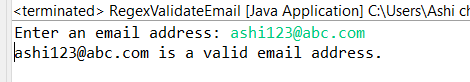
}

sc.close();

}

}

**OUTPUT:**

****

**8.     Create a Java program to validate phone numbers. The format should be (xxx) xxx-xxxx where x is a digit.**

**PROGRAM:**

**package** Lab6;

**import** java.util.Scanner;

**import** java.util.regex.Pattern;

**public** **class** RegexValidatePhoneNumber {

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

// Regex for phone number

String phoneRegex = "^\\(\\d{3}\\) \\d{3}-\\d{4}$";

// Compiling the regex

Pattern p = Pattern.*compile*(phoneRegex);

Scanner sc = **new** Scanner(System.***in***);

// taking user input for phone number using scanner object

System.***out***.print("Enter a phone number in the format (xxx) xxx-xxxx: ");

String phoneNumber = sc.nextLine();

// Boolean variable to check if phone number matches the regex pattern

**boolean** isValid = p.matcher(phoneNumber).matches();

// Printing valid or invlaid based on boolean variable pattern

**if** (isValid) {

System.***out***.println(phoneNumber + " is a valid phone number.");

} **else** {

System.***out***.println(phoneNumber + " is not a valid phone number.");

}

sc.close();

}

}

**OUTPUT:**

