

## Assignment 1 (Unit-1)

Q.1) Write a Java program to check whether the number is prime or not.

→

Code :

```
import java.util.Scanner;
```

class Q1 {

```
    public static boolean isPrime(int no) {
```

```
        for (int i = 2; i < (no / 2 + 1); i++) {
```

```
            if (no % i == 0)
```

```
                return false;
```

}

```
        return true;
```

} // end of for loop

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

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

```
        int no = sc.nextInt();
```

```
        boolean result = isPrime (no);
```

if (result) System.out.println ("The number " + no + " is prime");

```
        else System.out.println ("The number " + no + " is not prime");
```

}

} // end of class Q1

(2-3) & functions

Output: : shows output of a file

CMD

javac Q1.java

java Q1

: abcd

Enter a number : 37

The number 37 is prime

(on int) with random static fields

3++i : (i+s) [CMD] i = 1, s = 3

(o = i + ron) 31

java Q1

Enter a number : 6

The number 6 is not prime.

Q2. Write a java program to generate a ladder of numbers.



Code: : main() is static method

import java.util.Scanner;

"pi" <--> "random set" string -> integer

((String don) : class Q2 { } )

public static void main(String[] args) {

To print a box Scanner sc = new Scanner(System.in);

No. of rows set System.out.print("Enter number of rows : ");

int n = sc.nextInt();

for(int i=1; i<=n; i++) {

    for(int j=1; j<=i; j++) {

        System.out.print(j + " ");

}

    System.out.println();

}

Copy [Print] from the slide in a single

3

:(i,message) number n = 5

Output: java Q2.java

CMD : java

java Q2.java

Enter number of rows : 5

1 2 3 4 5

1 2 3 4 5

1 2 3 4 5

1 2 3 4 5

? (Question mark)

Ques: If the following segment : qmst-fai

: quest("m/f?")-format: fun, anskey?

: qmst->m/f

?

Q3) i. Write a java programs which will read a line of integers, and displays each integer and the sum of all integers using StringTokenizer.



Code:

```
import java.util.Scanner;  
import java.util.StringTokenizer;  
  
class Q3 {  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in);  
        System.out.print("Input number to add in single  
line");  
        String str = sc.nextLine();  
        StringTokenizer st = new StringTokenizer(str);  
        int sum = 0;  
  
        System.out.println("Input : ");  
  
        while (st.hasMoreTokens()) {  
            int temp = Integer.parseInt(st.nextToken());  
            System.out.format("%d\n", temp);  
            sum += temp;  
        }  
    }  
}
```

```
: private void sum() { System.out.println("Sum of numbers : " + sum); }
```

points ③ Enclosed in a stratum.  
points ③ Enclosed in a stratum.

### Output:

**CMD** 22:54:00? /H. pagi - 300 mi

javac Q.3.java

java Q3

~~Input numbers to add in a single~~

line : 16 32 64 128 256 512 1024 2048

Input: 300 300

~~difficult, hard~~ = difficult, hard

~~(Answer to "What do others" 32 thing, does, make?)~~

128 : (" von H. D.)

256 (cont'd)

(n " + (300) 512 *g. f. u. m. t. g.*

1024

2048

Sum of numbers : 4080

3 C " 1 model

Q.4) Write a program to calculate the following:

1. Find the length of array.
2. Demonstrate a one-dimensional array
3. Demonstrate a Two dimensional array.



Code:

```
import java.util.Scanner;
```

```
class Q4 {
```

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

```
int [] arr = {15, 30, 45, 60, 75, 90, 105, 120};
```

```
int length = arr.length;
```

```
System.out.println("Length of array " + length);
```

```
System.out.println("Demonstrate one-dimensional  
array");
```

```
for (int i=0; i<length; i++) {
```

```
System.out.print (arr[i] + " ");
```

```
}
```

```
System.out.println("Demonstrating two dimensional  
array");
```

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

```
System.out.println("Enter number of rows : ");
```

```
int rows = sc.nextInt();
```

```
System.out.print("Enter number of columns : ");
```

```
int column = sc.nextInt();
```

```
int[][] matrix = new int[rows][column];
```

```
for(int i = 0; i < rows; i++) {
```

```
: for(int j=0; j<column; j++) {
```

```
System.out.print ("Enter element at position  
[" + (i+1) + "][" + (j+1) + "] : ");
```

```
matrix[i][j] = sc.nextInt();
```

3

3

```
System.out.println("Printing Two-dimensional array");
```

```
for(int i=0 ; i<rows ; i++) {
```

```
for(int j=0; j < columns; j++) {
```

```
System.out.print("matrix[i][j] + " ");
```

3) *advice to generate code*

```
: System.out.println();
```

3. (c) Setting the standards

z : 3 [exes] addition to strings defined

3: 6353 oddane do atomskie stany

5 : [Editor] options to download media

8 : [3] [3] index du Journal 2013

E : (e3 E3 collision de structures dans)

Output : [CMD] javac P4.java & java P4

javac P4.java \$12xen,28 = success fri

java P4 [array] fri sun = xisten [31] fri

length of array :: 8 or 21 : 0 = 31 fri

Demonstrate one dimensional array:

printing from 15 to 120

i (" : [" + (i+1) + "][ " + (i+1) + "]")

Demonstrating two-dimensional array

Enter two dim

Enter no. of rows :

3

Enter number of columns :

3

3

Enter elements at position [2][1] : 1

Enter elements at position [1][2] : 2

Enter elements at position [2][3] : 3

Enter elements at position [2][2] : 4

Enter elements at position [2][1] : 5

Enter elements at position [2][3] : 6

Enter elements at position [3][1] : 7

Enter elements at position [3][2] : 8

Enter elements at position [3][3] : 9

( " : word list will ) forming , then , method

Printing two dimensional array:

( (1, 2, 3) forming , then , method )

4 5 6

7 8 9

( (row - wise ) form )

Q5) Write a java program to display strings in sorted order.

( ( ++ i + first , second , i = 0 = i do ) )

Code: ( (1) Java , " n/e/c" ) forming , then , method

```
import java.util.Scanner;
```

```
import java.util.Arrays;
```

class Q5 {

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

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

```
    System.out.println("Enter five words : ");
```

```
    String[] arr = new String[5];
```

```
    for (int i=0; i<5; i++) {
```

```
        System.out.print("Enter word no. " + (i+1))
```

```
        + " : ");
```

```
    arr[i] = sc.next();
```

}

```
System.out.println("Unsorted Array :");  
for( int i = 0; i < arr.length; i++ ) {  
    System.out.format("%s\n", arr[i]);  
}  
  
Arrays.sort(arr);
```

```
System.out.println("Sorted Array :");  
for( int i = 0; i < arr.length; i++ ) {  
    System.out.format("%s\n", arr[i]);  
}  
3  
3  
3
```

Output:

CMD

Enter five words: India, China, USA, UK, Australia

Enter word no. 1 : India

Enter word no. 2 : Australia

Enter word no. 3 : England

Enter word no. 4 : America

Enter word no. 5 : Brazil

Unsorted Array:

India

Australia

England

America

Brazil

Sorted Array:

America

Australia

Brazil

England

India

Q.6) Explain in detail Scanner class, BufferedReader and  
Console Class with all its methods and usage.

Scanner Class:

- The Scanner class is used to get user input, and it is found in the java.util package.
- To user the Scanner class, created an object of the class and use any of the available methods found in the Scanner class documentation.
- Various different functions of Scanner class are:

• next()

• nextLine()

• nextInt()

- nextLong()
- nextFloat()
- nextDouble()

- next() - It finds and returns the next complete token from the Scanner.
- nextLine() - It moves the Scanner position to next Line and returns input as a string.
- nextInt() - It scans the next token of input as integer.
- nextLong() - It scans the next token of input as Long.
- nextFloat() - It scans the next token of input as float.
- nextDouble() - It scans the next token of input as double.

## 2) BufferedReader class:

- The java.io package contains BufferedReader class that takes input as an argument that is an object of InputStreamReader .
- For input the function - readLine() is used .
- This readLine function might generate an I/O Exception . So , we write it in try catch block .

```
InputStreamReader isr = new InputStreamReader
( System.in );
```

(S find) so, in InputStreamReader

BufferedReader br = new BufferedReader (isr);

3) Use System.Console class:

(iii) ~~multiple~~ : The console class is used to get input from user and also to display output.~~multiple~~ It provides methods to read texts and password. If you read password using Console class, it will not be displayed to the user.

- Syntax,

String name = System.Console.ReadLine();

String password = System.Console.ReadPassword();

- The console is used to get input from the command line prompt.

System.out.println("Enter Name");

System.out.println("Enter Password");

System.out.println("Enter Password");

(navigated) rename war : 02 rename.html

File &gt; Save As &gt; bior

File &gt; Save As &gt; bior

{}

File &gt; Save As &gt; bior

File &gt; Save As &gt; bior

File &gt; Save As &gt; bior