PROGRAM:1

```
import java.io.*;
public class Test {
 public static void main(String[] args)
 throws IOException
 {
 File file = new File("C:\\Users\\HP\\Desktop\\text.txt");
 FileInputStream fileInputStream = new FileInputStream(file);
 InputStreamReader inputStreamReader = new
InputStreamReader(fileInputStream);
 BufferedReader bufferedReader = new BufferedReader(inputStreamReader);
 String line;
 int wordCount = 0;
 int characterCount = 0;
 int paraCount = 0;
 int whiteSpaceCount = 0;
 int sentenceCount = 0;
 while ((line = bufferedReader.readLine()) != null) {
 if (line.equals("")) {
 paraCount += 1;
 }
 else {
 characterCount += line.length();
 String words[] = line.split("\\s+");
 wordCount += words.length;
 whiteSpaceCount += wordCount - 1;
 String sentence[] = line.split("[!?.:]+");
 sentenceCount += sentence.length;
 }
 }
 if (sentenceCount >= 1) {
 paraCount++;
```

```
}
System.out.println("Total word count = "+ wordCount);
System.out.println("Total number of sentences = "+ sentenceCount);
System.out.println("Total number of characters = "+ characterCount);
System.out.println("Number of paragraphs = "+ paraCount);
System.out.println("Total number of whitespaces = "+ whiteSpaceCount);
}
```

```
Total word count = 10
Total number of sentences = 2
Total number of characters = 49
Number of paragraphs = 1
Total number of whitespaces = 13
```

PROGRAM:2

```
class Customer{
 int amount=10000;
 synchronized void withdraw(int amount)
 {
 System.out.println("going to withdraw...");
 if(this.amount<amount)
 {
 System.out.println("Less balance; waiting for deposit...");  try
 {
 wait();
 }
 catch(Exception e)
 {}
 }
 this.amount-=amount;
 System.out.println("withdraw completed...");
 System.out.println(amount);
 }
 synchronized void deposit(int amount)
 {
```

```
 System.out.println("going to deposit...");
 this.amount+=amount;
 System.out.println("deposit completed... ");
 notify();
 }
}
class TestIC{
 public static void main(String args[]){
 final Customer c=new Customer();
 new Thread()
 {
 public void run()
 {
 c.withdraw(15000);
 }
 }.start();
 new Thread()
 {
 public void run()
 {
 c.deposit(10000);
 }
 }.start();
 }
}
```

```
going to withdraw...
Less balance; waiting for deposit...
going to deposit...
deposit completed...
withdraw completed...
15000
```

PROGRAM:3

```
import java.util.*;
class fizz
{
public static void main(String[] args)
{
try
 {
int n;
Scanner sc=new Scanner(System.in);
System.out.println("Enter the value");
n=sc.nextInt();
if (n%5==0 && n%3==0)
System.out.println("FizzBuzz");
else if (n%5==0)
System.out.println("Buzz");
else if (n%3==0)
System.out.println("Fizz");
else
System.out.println("Enter a number divisible by 3 or 5");
}
catch(Exception e)
 {
System.out.println("Due to character exception");
 }
}
}
```

Program 4

```
import java.io.*;
import java.util.*;
public class Solution
{
 public static void main(String[] args) {
 String str1, str2;
 Scanner sc =new Scanner(System.in);
 System.out.println("Enter the first string : ");
 str1=sc.nextLine();
 System.out.println("Enter the Second string : ");
 str2=sc.nextLine();
 if(str1.length() != str2.length()){
 System.out.println("Second string is not a rotation of first string");
 }
 else {
 str1 = str1.concat(str1);
 if(str1.indexOf(str2) != -1)
 System.out.println("Second string is a rotation of first string");
 else
 System.out.println("Second string is not a rotation of first string");
 }
 }
}
```

```
Enter the first string :
abcde
Enter the Second string :
cdeab
Second string is a rotation of first string
```

PROGRAM:5

```
class PrimeExample implements Runnable
//extends Thread
{
 public void run()
 {
 int i,m=20,flag=1;

 for(i=1;i<=m;i++)
 {
 if(i<=3){
 System.out.println(i + " is prime number");
 continue;
 }
 else if(i>3)
 {
 for(int j=2;j<i;j++)
 {
 if(i%j==0)
 {
 flag=0;
 break;
 }
 }
 if (flag!= 1)
 {
 System.out.println(i + " is not prime number");
 flag=1;
 }
 else
 System.out.println(i + " is prime number");
 }
```

```
 }
 }
}
class prime
{
public static void main(String args[]){
 try
 {
 PrimeExample p1 = new PrimeExample();
 Thread t1= new Thread(p1);
 t1.start();
 }
 catch(Exception e)
 {
 System.out.println(e.getMessage());
 }
}
}
```