4. WAP to check if a number is palindrome

```
class Palindrome {
    public static void main (String args[]){
        int a=12121 , ans=0, rem;
        int b=a;
        while (a!=0){
            rem=a%10;
            ans= ans*10+rem;
            a=a/10;
        }
        if (ans==b) {
            SOP("No is pallindrome");
        }
        else {
            SOP ("No is not a pallindrome");
        }
    }
}
```

O/P

Number is pallindrome

classmate

Date
Page

22.12.2023 (1)

```
class Overload {

    void print(int n){
        int sum = 0;
        for (int i = 1; i<=n; i++) {
            sum = sum + i;
        }
        System.out.println("Sum of "+n+" natural
            numbers is "+sum);
    }

    void print (int m, int n) {

        System.out.println("Prime numbers in the
            range are");
        for (int i = m; i <= n; i++){
            int flag = 0;
            for (int j = 2; j <= i/2; j++){
                if (i % j == 0) {
                    flag = 1;
                    break;
                }
            }
            if (flag == 0)
                System.out.println(i);
        }
    }
}

class OverloadDemo {

    public static void main (String [] args) {
        Overload o = new Overload();
        o.print(5);
        o.print(7,13);
    }
```

}

Output

sum of 5 natural numbers is 15

Prime numbers in the range are 2, 3, 5, 7, 11, 13

2. class Grocery {

String c_name;

String c_ph;

double total;

Grocery (String c_name, String c_ph){

this.c_name = c_name;

this.c_ph = c_ph;

}

void calc (double q_dal, double q_pulses, double q_sugar){

total = q_dal*100 + q_pulses*80 + q_sugar*50;

}

void display()

{

System.out.println("Name " + " " + " Phone number " + " " + " Total");

System.out.println(c_name + " " + c_ph + " " + total);

System.out.println();

}

}

class GDemo {

public static void main(String [] args){

Grocery g1 = new Grocery("Rama", "8060302010");

Grocery g2 = new Grocery("Bhama", "7689632510");

Grocery g3 = new Grocery("Bhama", "9632587412")

```
g1.calc(2,2,2);
g2.display();

g2.calc(3,5,2);
g2.display();

g3.calc(1,1,0.5);
g3.display();

}
}
```

output :-

Name	Phone number	Total
Ram	8060302010	410.0

Program 3:-

```
③ import java.util.Scanner;

class Quad {
    int a,b,c;
    double r1,r2,d;
    Scanner s=new Scanner(System.in);

    void input()
    {
        System.out.println("Quadratic equation is in the
                                form: ax^2+bx+c");
        System.out.print("Enter a:");
        a = s.nextInt();
        System.out.print("Enter b:");
        b = s.nextInt();
        System.out.print("Enter c:");
        c = s.nextInt();
    }
    void calculateRoots(){
        if(d>0)
        {
```

```
System.out.println("Roots are real and unequal");
root1 = (-b + Math.sqrt(d))/(2*a);
root2 = (-b - Math.sqrt(d))/(2*a);
System.out.println("First root is:" + root1);
System.out.println("Second root is:" + root2);
}

else if(d==0)
{
System.out.println("Roots are real and equal");
root1 = (-b + Math.sqrt(d))/(2*a);
System.out.println("Root:" + root1);
}
else
{
System.out.println("No real solution. Roots are imaginary");
double real = -b/(2*a);
double imaginary = Math.sqrt(-d)/(2*a);
System.out.println("The equation has 2 complex roots:" +
real + "+" + imaginary + "i and" + real + "-" + imaginary +
"i");
}
}
}
class Main {
public static void main (String [] args){
Quad q = new Quad();
q.input();
q.discriminant();
q.calculateRoots();
```

Output:-

Enter quadratic eqn in form of $ax^2 + bx + c$

Enter $a = 2$

Enter b = 2
Enter c = 2
Roots are imaginary
First root is = 0.0 + i 0.86602
Second root is = 0.0 - i 0.86602

22-12-23

2024

Lab-2

Q. Create a class Book that contains 4 members: name, author, price, num_pages. Include a constructor to set the values for the members. Include methods to set and get the details of the object. Include toString() method that could display the complete details of the book. Develop a Java program to create n Book objects.

```
import java.util.Scanner;

class Book
{
    String name;
    String author;
    int price;
    int numPages;

    Book() {}
    Book(String name, String author, int price, int numPages)
    {
     this.name = name;
     this.author = author;
     this.price = price;
     this.numPages = numPages;
    }
    public String toString()
    {
      String name, author, price, numPages;
      name = "Book name:" + this.name + "\n";
      author = "Author name:" + this.author + "\n";
      price = "Price :" + this.price + "\n";
      numPages = "number of pages:" + this.numPages + "\n";

      return name + author + price + numPages;
    }
}
```

```
class Main
{
    public static void main (String args[])
    {
     Scanner s = new Scanner (System.in);
     int n;
     String name;
     String author;
     int price;
     int numPages;
     System.out.println("Enter the no of books:");
     n = s.nextInt();
     Book b[];
     b = new Book [n];

     for (int i = 0; i < n; i++)
     {
      System.out.println("Book" + (i+1) + ":");
      System.out.print("Enter name of book:");
      name = s.next();
      System.out.print("Enter author:");
      author = s.next();
      System.out.print("Enter price:");
      price = s.nextInt();
      System.out.print("Enter no of pages:");
      numPages = s.nextInt();
      b[i] = new Book (name, author, price, numPages);
     }
     for (int i = 0; i < n; i++)
      System.out.print("Book" + (i+1) + ":\n" + b[i]);
    }
}
```

Output:-

Enter the number of book: 1

Book 1 :-

Enter the name of the Book :

~~Ramayan~~ abc

Enter the author of the book:

~~Honey Singh~~ cde

Enter the price of the book:

1500

Enter the number of pages of the book:

150

Book 1 :-

Book name: ~~Ramayan~~ abc

Author : ~~Honey Singh~~ cde

Price : 1500

~~Pag~~ Number of Pages : 150

(a) Write a Java program to create a class Student with members USN, name, ~~marks~~ (6 subjects). Include methods to accept Student details and marks. Also include a method to calculate the percentage and display appropriate details. (Array of Student object to be created)

```
import java.util.Scanner;
class Student {
    String usn;
    String name;
    int marks[] = new int[6];

    void Detail ()
    {
        Scanner s = new Scanner(System.in);
```

```
System.out.println("Enter USN");
USN = s.next();
System.out.println("Enter name");
name = s.next();
System.out.println("Enter marks for 6 subjects:");
for (int i=0; i<6; i++)
{
    system.out.print("Subject "+(i+1)+":");
    marks[i]=s.nextInt();
}
}

double percentage()
{
int total =0;
for (int i=0; i<6; i++)
{
  total += marks[i];
}
double p = total /6;
return p;
}
void display()
{
  System.out.println("USN:"+USN);
  System.out.println("Name:"+name);
  System.out.println("Marks:");
  for (int i =0; i<6; i++)
  {
  System.out.println("Subject"+(i+1)+":"+marks[i]);
  }
  System.out.println("Percentage:"+percentage()+"%");
}
}
```

```
class Lab2Student
{
  public static void main(String args[]){
    Scanner s = new Scanner(System.in);
    System.out.print("Enter the number of students :");
    int n = s.nextInt();
    Student[] students = new Student[n];
    for (int i=0; i<n; i++)
    {
      students[i] = new Student();
      System.out.println("\n Enter details for student " + (i+1) + ":");
      students[i].Details();
    }
    for (Student student : students)
    {
      student.display();
    }
  }
}
```

12-1-24

Output :-

```
Enter the number of students : 1
Enter details for Student 1:
Enter USN
1234
Enter name
nomo
Enter marks for 6 subjects:-
Subject 1:67
Subject 2: 89
Subject 3:99
Subject 4:77
Subject 5:80
Subject 6:80
```

Student Details :

USN : 1234 Name : momo

Marks :

Subject 1 : 67

Subject 2 : 89

Subject 3 : 99

Subject 4 : 77

Subject 5 : 80

Subject 6 : 80

Percentage : 82.0%

Lab 4

1. Develop a Java program to create an abstract class named shape that contains two integers and an empty method named printArea(). Provide 3 classes named Rectangle, Triangle and Circle such that each one of the classes extends the class shape. Each one of the classes contains the method printArea() that prints the area of the given shape.

Implat :-

→ create an abstract class shape
→ Initialise 2 integers a and b
→ Declare an empty method named printArea()
→ create class rectangle extends shape
→ then create super class and initialise value of a and b
→ Declare method printArea1()
→ Create class Triangle extends shape
→ Declare method printArea2()
→ Create class Circle extends shape
→ Declare method printArea3()
→ Create class main and call back all methods while initialising an object shape.
→ Enter values of a and b for each of them.

code:-
```
abstract class Shape {
    int a;
    int b;
    abstract void printArea();
}
class Rectangle extends Shape {
    Rectangle (int length, int breath) {
      a = length;
      b = breath;
    }
    void printArea(){
        System.out.println("area of rectangle is " + a*b);
    }
}
```

```
class Triangle extends Shape {
        Triangle (int length, int height) {
                a = length ;
                b = height ;
        }
        void printArea(){
                System.out.println("Area of Triangle is" + a*b/2);
        }
}

class Circle extends Shape {
        Circle (int radius, int not){
        a = radius ;
        b = not ;
        }
        void printArea() {
                System.out.println("Area of Circle is" + 3.14*a*a);
        }
}

class Run {
        public static void main (String args[]) {
        Rectangle r1 = new Rectangle (9,6);
        Triangle t1 = new Triangle (9,6);
        Circle c1 = new Circle (2,3);

        r1.printArea();
        t1.printArea();
        c1.printArea();
        }
}
```

8/1/24

Output :-

Area of rectangle is 24
Area of triangle is 27
Area of circle is 12.56.

Bank account :-

```
import java.ut
class Account
{ String nam
  int acc
  String
  doub

  Account (a

  { this.
    this
    this
    this
  }
  void
  {
  }
  void
  {
  }
}
```

2. Bank account :-

```
import java.util.Scanner;
class Account
{ String name;
    int acc no;
    String type;
    double balance;

    Account (String name, int acc no, String type, double
            balance)
    { this.name = name;
      this.acc no = acc no;
      this.type = type;
      this.balance = balance;
    }
    void deposit (double amount)
    { balance += amount;
    }
    void withdraw (double amount)
    {
        if ((balance - amount) >= 0)
        {
            balance -= amount;
        }
        else
        {
            System.out.println("Insufficient balance can't
                                withdraw");
        }
    }
    void display()
    { System.out.println ("name:" + name + "acc no" + acc no
                          + "type:" + type + "balance" +
                          balance);
    }
}
```

```
class SavAcct extends Account
{
    private static double rate = 5;

    SavAcct (String name, int acc_no, double balance)
    { super (name, acc_no, "saving", balance);
    }
    void interest ()
    { balance += (balance * rate)/100;
      System.out.println("Balance" + balance);
    }
}
class CurAcct extends Account
{
    private double minBal = 500;
    private double service charge = 50;

    CurAcct (String name, int acc_no, double balance)
    {
     super (name, acc_no, "current", balance);
    }
    void check Min()
    { if (balance < minBal)
        {
          System.out.println("balance is less than min
                balance, service charge imposed" + service
                charge);
          System.out.println("balance is:" + balance)
        }
    }
}

class account Main()
{
```

```
public static void main (String a[])
{
 Scanner s = new Scanner (System.in);
 System.out.println("Enter name");
String name = s.next();
  System.out.println("Enter the account type:");
String type = s.next();
 System.out.println("Enter acc number:");
 int acc no = s.nextInt();
 System.out.println("Enter initial balance");
  double balance = s.nextDouble();
  int ch;
  double amount 1, amount 2;
  Account acc = new Account(name, acc no, type, balance);
 SavAcct sa = new SavAcct(name, acc no, balance);
  CurAcct ca = new CurAcct(name, acc no, balance);
  while (true)
  {  if (acc.type.equals("savings"))
  {
      System.out.println("\n menu\n\ 1. Deposit 2. withdraw
                                 3. compute interest 4. Display");

      System.out.println("Enter the choice:");
       ch = s.nextInt();

  switch (ch)
  { case 1: system.out.println("Enter the amount");
     amount 1 = s.nextInt();
      sa.deposit (amount 1);
      break;

   case 2: system.out.println("Enter the amount:");
     amount 2 = s.nextInt();
     sa.withdraw (amount 2);
     break;
```

```
case3 : sa.withdraw()
break;

case4: sa.display();
break;

case5 : System.exit(0);
default : System.out.println("Invalid input");
break;
}
}
else
{System.out.println("\n menu \n 1. deposit 2. withdraw
                    3. display");

System.out.println("Enter the choice");
ch = s.nextInt();
switch(ch)
{
case1 : System.out.println("Enter the amount");
amount1 = s.nextInt()
ca.deposit(amount1);
break;

case 2: System.out.println("Enter the amount :");
amount2 = s.nextInt();
ca.withdraw(amount2);
ca.checkmin();
break;

case3: ca.display();
break;
case4 : system.exit(0);
default : System.out.println("Invalid input");
break;
}
```

```
        }
    }
}
```

//output :-

Enter the name: Sohan
Enter the type: current;
Enter the account Number;
12104525
Enter the initial balance
15000
Menu
1. Deposit 2. withdraw 3. Display
Enter the choice
3
name: Sohan Acc no: 12104525 type: current
balance: 14800

(Q) Write a program that demonstrate handling of exception in inheritance then, create a base class called "Father" and derived class called "Son" which extends the base class. In Father class, implement a constructor which takes the age and throw the exception wrongAge() when the input age <0. In Son class, implement a constructor which ~~takes the~~ takes both father and son's age and throw an exception if son's age is >= father's age.

Program:-

```
import java.util.Scanner;
  Public class Exception Inheritance
{
    Static class Father
    { int fage;
```

```
Father (int fage) throws WrongAge {
    if (age<0)
    {throw new WrongAge("Father's age cannot
    be negative.");
    }
    this.fage = fage;
  }
}

static class son extends Father
  {int sage;
  son(int fage, int sage) throws WrongAge
    {
      if(sage >= fage)
      {
        throw new WrongAge("son's age
          cannot be equal or greater than father's
          age");
      }
      this.fage = fage;
      this.sage = sage;
    }
}

static class WrongAge extends Exception
{
  WrongAge(String message)
    {
      super(message);
    }
}

public static void main (String args[])
{
  Scanner s = new Scanner(System.in);
  System.out.println("Enter father's age");
  int fatherAge = s.nextInt();
```

```
System.out.println("Enter Son's Age:");
int SonAge = s.nextInt();

try {
    Father f = new Father(fatherAge);
    Son son = new Son(fatherAge, SonAge);
    System.out.println("Father's Age:" + fatherAge);
    System.out.println("Son's age:" + SonAge);
}
catch (WrongAge e)
{
    System.out.println(e.getMessage());
}
}
}
```

//output :-

Father's age : 50

Son's age : 25

Q) Create a package CIE which has 2 classes - Student and Internals. The class Student has members like usn, name, sem. The class internals derived from student has array that stores the internal marks scored in 5 courses of the current semester of the students. Create another package SEE which has the class External which is a derived class of student. This class has an array that stores the SEE marks scored in five courses of the current semester of the student. Import the 2 packages in a file that declares the final marks of n students in all five courses.

1. Create a folder CIE and save the programs Student.java and Internals.java within it.

2. Create a folder SEE and save the program External.java

3. Save the main program outside these two folders.

4. Compile Main.Java and Execute the Main.class.

```
Package SEE;
import CIE.Internals;
import java.util.Scanner;

public class Externals extends Internals {
        protected int marks[];
        protected int finalMarks[];

        public External ){
            marks= new int [5];
            finalMarks = new int [5];
        }
```

classes - Student
members like
d from student
cored in 5 courses
Create another pack
in a desired dom
stores the SEE
current semester of th
file that declares th
course.

program Student.java

folder.

main.class -

```
~~package SEE~~ public void inputSEEmarks(){
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter SEE Marks for" + name);
        for (int i=0; i<5; i++){
            System.out.print("Subject" + (i+1) + "marks:");
            marks[i] = scanner.nextInt();
        }
    }
    public void calculateFinalMarks(){
        for (int i=0; i<5; i++)
            finalMarks[i] = marks[i]/2 + ~~[illegible]~~
                                marks[i];
    }
    public void displayFinalMarks(){
        displayStudentDetails();
        for (int i=0; i<5; i++)
            System.out.println("Subject" + (i+1) +
                ":" + finalMarks[i]);
    }
}

package CIE;
import java.util.Scanner;

public class Internals extends Student {
    protected int marks[] = new int[5];

    public Internals(){
    }
    public void inputCIEmarks(){
        Scanner scanner = new Scanner(System.in);
        System.out.println("Enter internal marks for" + name);
        for (int i=0; i<5; i++){
            System.out.print("Subject" + (i+1) + "marks:");
            marks[i] = scanner.nextInt();
        }
    }
}
```

```
import SEE. External;
public class main {
    public static void main( String args[]) {
        int numOf students = 2;
        External finalMarks[] = new External
                                [numOf students];
        for (int i=0; i < numOf students; i++){

            finalMarks[ i ]=new External( );
            finalMarks[ i ]. inputStudentDetails();
            System. out. println ("Enter CIE Marks");
            finalMarks[i] . inputCIEmarks();
            System. out. println ("Enter SEE marks");
            finalMarks[ i ]. inputSEE marks();
        }

        System. out. println("Displaying Data:\n");

        for (int i=0; i<numOf Students; i++) {
            finalMarks [i]. calculateFinalMarks();
            finalMarks [i]. displayFinalMarks();
        }
    }
}


package CIE;
import java. util. Scanner;
public class Student {
    protected String usn= new String ();
    protected String name= new String();
    protected int sem;

    public void inputStudentDetails(){
        Scanner scanner= new Scanner (System. in);
        System. out. println("Enter USN:");
```

```
usn = scanner.next();
System.out.println("Enter name:");
name=scanner.next();
System.out.print("Enter semester:");
sem= scanner.nextInt();
}
public void displayStudentDetails(){
        System.out.println("USN:"+usn);
        System.out.println("Name:"+name);
        System.out.println("Semester:"+sem);
    }
}
```

Output :-

```
Enter USN :1BM22CS314
Enter name: Vachan
Enter Semester: 2
Enter CIE marks:
Enter internal marks for Vachan
Subject 1 marks: 48
Subject 2 marks: 47
Subject 3 marks: 46
Subject 4 marks: 49
Subject 5 marks: 50
Enter SEE marks:-
Enter SEE marks for Vachan
Subject 1 marks : 45
Subject 2 marks : 43
Subject 3 marks : 42
Subject 4 marks : 48
Subject 5 marks : 49

Enter USN: 1BM22CS291
Enter Name: Sudu
Enter Semester: 2
Enter CIE marks:
Enter internal marks for Sudu
Subject 1 marks: 48
Subject 2 marks: 49
Subject 3 marks 48
Subject 4 marks: 48
Subject 5 marks: 50
Enter SEE marks:-
Enter SEE marks for Sudu
Subject 1 marks: 45
Subject 2 marks : 42
Subject 3 marks : 43
Subject 4 marks: 45
Subject 5 marks : 43

Displaying data:
USN: 1BM22CS314
Name: Vachan
Semester: 2
Subject 1: 70
Subject 2: 68
Subject 3: 67
Subject 4: 72
Subject 5: 74
```

USN: 1BM22CS300
Name: Sudu
Semester: 2 Subject 1: 70 Subject 2: 70 Subject 3: 70 Subject 4: 71 Subject 5: 71

(6a) Write a program which creates two threads one thread displaying "BMS college of Engineering" once every ten seconds and another displaying "CSE" once every two seconds.

```
class DisplayThread extends Thread {
        private final String message;
        private final long interval;

        DisplayThread( String message, long interval)
        {
            this.message = message;
            this.interval = interval;
        }
        public void run()
        { try
            { while (true)
                { System.out.println(message);
                  Thread.sleep(interval);
                }
            }
            catch (InterruptedException e)
            { System.out.println(Thread.currentThread
              .getName()+" interrupted");
            }
        }
    }
public class TwoThreadDemo {
        public static void main (String args[]) {
            DisplayThread t1 = new DisplayThread("BMS
                College of Engineering", 1000);

            DisplayThread t2 = new DisplayThread("CSE", 2000);
```

```
t1.setName("Thread1");
t2.setName("Thread2");

t1.start();
t2.start();

try {
    Thread.sleep(3000);
  }
catch (InterruptedException e)
{
  System.out.println("main Thread Interrupted");
  }
    t1.interrupt();
    t2.interrupt();

    System.out.println("main thread exiting");
  }
}
```

// Output :-

```
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
CSE
CSE
CSE
CSE
CSE
CSE
CSE
CSE
CSE
BMS College of Engineering
CSE
```

Date
Page

(21) Creating label, Button and Text Field in a Frame using AWT.

```
import java.awt.*;
import java.awt.event.*;

public class AWTExample extends WindowAdapter {
    Frame f;
    AWTExample(){
        f = new Frame();
        f.addWindowListener(this);
        Label l = new Label("Employee id:");
        Button b = new Button("Submit");
        TextField t = new TextField();

        l.setBounds(20,80,80,30);
        t.setBounds(20,100,80,30);
        b.setBounds(100,100,80,30);

        f.add(b);
        f.add(l);
        f.add(t);

        f.setSize(400,300);
        f.setTitle("Employee info");
        f.setLayout(null);
        f.setVisible(true);
    }
    public void windowClosing(WindowEvent e){
        System.exit(0);
    }
    public static void main(String[] args) {
        AWTExample awt obj = new AWTExample();
    }
}
```

// output :-

Employee id :-

[ ] [Submit]

2. Create a button and add a action listener for Mouse click.

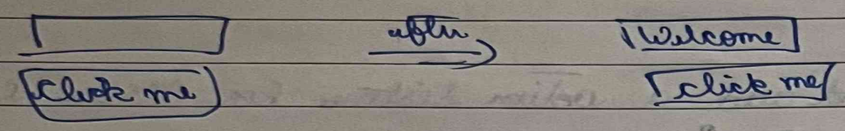
```
import java.awt.*;
import java.awt.event.*;
public class EventHandling extends WindowAdapter implements
 ActionListener {
        Frame f;
        TextField tf;
        EventHandling() {

            f = new Frame();
            f.addWindowListener(this);
            tf = new TextField();
            tf.setBounds(60,50,170,20);
            Button b = new Button("click me");
            b.setBounds(100,120,80,30);

            b.addActionListener(this);
            f.add(b); f.add(tf);
            f.setSize(300,300);
            f.setLayout(null);
            f.setVisible(true);
        }
        public void actionPerformed(ActionEvent e){
            tf.setText("Welcome");
        }
        public void windowClosing(WindowEvent e){
            System.exit(0);
        }
```

```
    public static void main (String args[]){
        new EventHandling();
    }
}
```

//output :-



Example 1 :-

```
import java.io.*;
public class ByteArrayInputS{
    public static void main (String [] args) throws
        IOException {
        byte [] buf={35,36,37,38};
        ByteArrayInputStream byt = new ByteArray
        InputStream (buf);
        int k=0;
        while ((k= byt.read()) != -1){
            char ch= (char)k;
            System.out.println("ASCII Value of
            character is:" +k+ "; Special character
            is:" + ch);
        }
    }
}
```

// output :-

```
ASCII value of character is:35; Special character is:#;
ASCII value of character is:36; Special character is:$;
ASCII value of character is:37; Special character is:%;
ASCII value of character is:38; Special character is:&
```

Example 2 :-

```
import java

public class
    publ
}
```

//output :-

Remaining Bytes
A 65 Remaini
Remaining bytes
Success .....

Example 3 :-

```
public class F
    publ
    File
    int
    Byt
```

Example 2 :-

```
import java.io.*;

public class ByteArray_ex {
    public static void main (String arg[]) Throws Exception {
        File OutputStream fout1 = new FileOutputStream ("Example.
                                txt");
        File OutputStream fout2 = new File OutputStream ("Example 3
          txt");

        ByteArrayOutputStream bout = new ByteArrayOutputStream();
        bout.write (65);
        bout.writeTo(fout1);
        bout.writeTo(fout2);

        bout.flush();
        bout.close();
        System.out.println ("Success.....");
    }
}
```

//output :-

~~Remaining Bytes that can be read: 1~~
~~A 65 Remaining bytes that can be read: 0~~
Remaining bytes that can be read: 0
Success .....

Example 3 :-

```
public class FileEx {
    public static void main (String a[]) throws IOException {
        File InputStream fin = new FileInputStream ("Example.txt");
        int content;
        System.out.println ("Remaining bytes that can be
            read:" + fin.available());
```

```
content = fin.read();
        System.out.print((char) content + "\n");
        System.out.print(content + " ");
        System.out.println("Remaining bytes that
        can be read:" + fin.available());

        System.out.println("Remaining bytes that
        can be read:" + fin.available());
    }
}
```

//output :-

Remaining Bytes that can be read :1
A 65 Remaining Bytes that can be read = 0
Remaining Bytes that can be read: 0

Example 4:-

```
import java.io.FileInputStream;
import java.io.IOException;

public class FileEx2E
    public static void main(String a[]) throws IOException
        {
        FileInputStream fin = new FileInputStream
                                ("Example.txt");

        byte[] bytes = new byte[20];
        int i;
        char c;
        i = fin.read(bytes);
        System.out.println("No of bytes read:"+i);
        System.out.print("Bytes read:");
        for (byte b: bytes){
            c = (char) b;
            System.out.print(c);
        }
}
```

// Output:-
Number of byte read : 1
Byte read : A