Lab Exercise on Files

1. **Write a program to create a text file in the path c:\java\abc.txt and check whether that file is exists or not.**

**CODE:**

import java.util.Scanner;

import java.io.\*;

public class q1{

public static void main(String[] args) throws IOException{

String path = "/home/student/java/abc.txt";

File f = new File(path);

if(f.createNewFile())

System.out.println("File has been created successfully!");

else

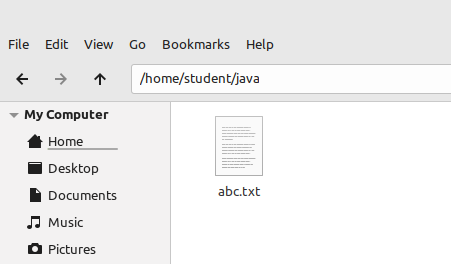
System.out.println("File is not created!");

}

}

**OUTPUT:**

****

****

1. **Write a program to accept specified number of characters as input, convert them into uppercase characters and save the data in a text file.**

**CODE:**

import java.util.Scanner;

import java.io.\*;

public class q2{

public static void main(String[] args) throws IOException{

Scanner in = new Scanner(System.in);

int i;

System.out.print("Enter no:of characters: ");

int n = in.nextInt();

char ch;

String path = "/home/student/java/abc.txt";

File f = new File(path);

if(f.exists())

{

FileWriter fw = new FileWriter(path);

for(i=0;i<n;i++)

{

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

ch=in.next().charAt(0);

fw.write(Character.toUpperCase(ch)+"\n");

}

System.out.println("All "+n+" characters have been written to the file!");

fw.close();

}

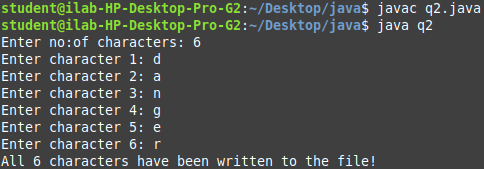
else

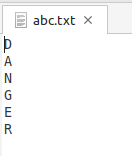
System.out.println("File is not found!");

}

}

**OUTPUT:**

****



1. **Write a program for a mail-order company. The program uses a data-entry screen in which the user types an item number and a quantity. The valid item numbers and prices are as follows:**

|  |  |
| --- | --- |
| Item-Number | Price |
| 101 | 100.50 |
| 105 | 175.95 |
| 108 | 220.45 |
| 115 | 280.75 |
| 125 | 300.99 |

**When the user enters an item number, check the number to make sure that it is valid. If it is valid, write the record that includes item number, quantity, price in a text file, otherwise your program should raise a user-defined exception.**

**CODE:**

import java.util.Scanner;

import java.io.\*;

class InvalidItemNoException extends Exception {

InvalidItemNoException(String s) {

super(s);

}

}

public class q3 {

public static boolean isValid(int iNo, int[] I, int m) throws InvalidItemNoException {

int flag = 0;

for (int i = 0; i < m; i++) {

if (iNo == I[i])

flag = 1;

}

if (flag == 0)

throw new InvalidItemNoException("Invalid Item No");

else

return true;

}

public static double price(int iNo, double[] p, int[] I, int m) {

for (int i = 0; i < m; i++) {

if (iNo == I[i])

return p[i];

}

return 0;

}

public static void main(String[] args) throws IOException {

Scanner in = new Scanner(System.in);

int i, m = 5, iNo, q;

int I[] = { 101, 105, 108, 115, 125 };

double P[] = { 100.50, 175.95, 220.45, 280.75, 300.99 };

String path = "C:\\java\\data.txt";

OutputStream os = new FileOutputStream(path);

DataOutputStream dos = new DataOutputStream(os);

System.out.print("Enter no:of records: ");

int n = in.nextInt();

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

try {

System.out.print("\nEnter Item no: ");

iNo = in.nextInt();

if (isValid(iNo, I, m)) {

System.out.print("Enter quantity: ");

q = in.nextInt();

dos.writeInt(iNo);

dos.writeInt(q);

dos.writeDouble(q \* price(iNo, P, I, m));

System.out.println("Record has been added!");

}

} catch (InvalidItemNoException e) {

System.out.println("Exception Occured: " + e);

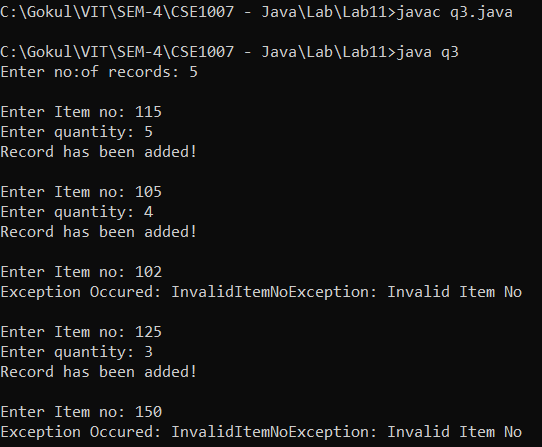
}

}

}

}

**OUTPUT:**



1. **Modify the above program such that, if the item number is valid, save the record in a file in object format, otherwise your program should raise a user-defined exception.**

**CODE:**

import java.util.Scanner;

import java.io.\*;

class Record implements Serializable{

private static final long serialVersionUID = 1L;

int iNo;

int q;

Double price, total;

Record(){};

Record(int iNo, int q,Double price, Double total)

{

this.iNo = iNo;

this.q = q;

this.price=price;

this.total = total;

}

public String toString()

{

return iNo + "\t" + q + "\t" + (total/q)+ "\t" + total;

}

}

class InvalidItemNoException extends Exception {

InvalidItemNoException(String s) {

super(s);

}

}

public class q4 {

public static boolean isValid(int iNo, int[] I, int m) throws InvalidItemNoException {

int flag = 0;

for (int i = 0; i < m; i++) {

if (iNo == I[i])

flag = 1;

}

if (flag == 0)

throw new InvalidItemNoException("Invalid Item No");

else

return true;

}

public static double price(int iNo, double[] p, int[] I, int m) {

for (int i = 0; i < m; i++) {

if (iNo == I[i])

return p[i];

}

return 0;

}

public static void main(String[] args) throws IOException {

Scanner in = new Scanner(System.in);

int i, m = 5, iNo;

int I[] = { 101, 105, 108, 115, 125 };

double P[] = { 100.50, 175.95, 220.45, 280.75, 300.99 };

String path = "C:\\java\\records.txt";

FileOutputStream f = new FileOutputStream(path);

ObjectOutputStream o = new ObjectOutputStream(f);

System.out.print("Enter no:of records: ");

int n = in.nextInt();

Record [] R = new Record[n];

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

try {

System.out.print("\nEnter Item no: ");

iNo = in.nextInt();

if (isValid(iNo, I, m)) {

R[i]=new Record();

System.out.print("Enter quantity: ");

R[i].q=in.nextInt();

R[i].iNo=iNo;

R[i].price=price(iNo, P, I, m);

R[i].total=R[i].q\*R[i].price;

o.writeObject(R[i]);

System.out.println("Record has been added!");

}

} catch (InvalidItemNoException e) {

System.out.println("Exception Occured: " + e);

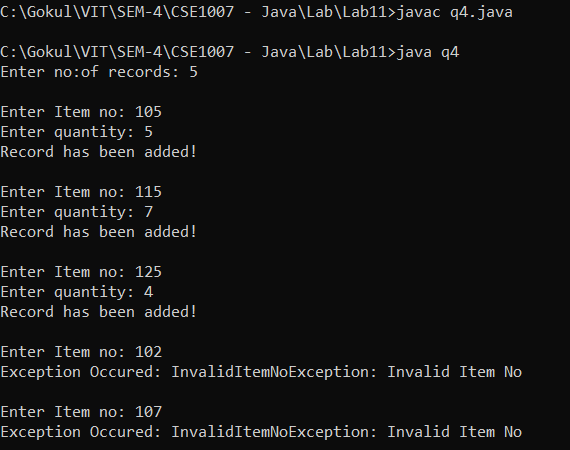
}

}

}

}

OUTPUT:



1. **Write a program that reads the data file created by the Q3 program and displays one record at a time on screen and finally prints the total price.**

**CODE:**

import java.util.Scanner;

import java.io.\*;

public class q5 {

public static void main(String[] args) throws IOException {

Scanner in = new Scanner(System.in);

int i, iNo, q;

double price, totalPrice, total=0;

String path = "C:\\java\\data.txt";

InputStream is = new FileInputStream(path);

DataInputStream dis = new DataInputStream(is);

System.out.println("Item No\tQty\tPrice\tTotal");

while(dis.available()>0){

iNo = dis.readInt();

q=dis.readInt();

totalPrice=dis.readDouble();

total+=totalPrice;

price = totalPrice/q;

System.out.println(iNo+"\t"+q+"\t"+price+"\t"+totalPrice);

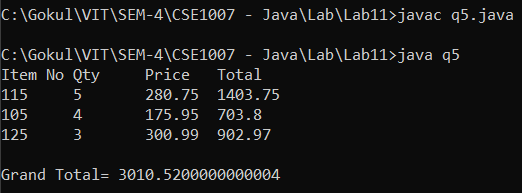
}

System.out.println("\nGrand Total= "+total);

}

}

**OUTPUT:**

****

1. **Consider a table “Employee” with fields: EmpID, EmpName, department and designation. Write a Java program to perform the following operations:**
   1. **Read employee name, department and designation and insert a new record. EmpID should be generated automatically such that new employee ID is one greater than maximum employee ID.**
   2. **Given an employee ID, your program should display other details.**
   3. **Given a department name, your program should print all employees belong to that department.**
   4. **Given a employee ID, your program should be able to change either her department or designation.**

**CODE:**

import java.util.Scanner;

import java.io.\*;

class Employee implements Serializable {

transient Scanner in = new Scanner(System.in);

public static int c = 101;

private static final long serialVersionUID = 1L;

int id;

String name, dept, desig;

Employee() {

id = c;

System.out.print("\nEnter name : ");

name = in.next();

System.out.print("Enter dept : ");

dept = in.next();

System.out.print("Enter desig: ");

desig = in.next();

c++;

};

public String toString() {

return id + "\t" + name + "\t" + dept + "\t" + desig;

}

}

public class q6 {

public static int eno(int ID, Employee[] Emp, int n){

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

{

if(ID==Emp[i].id)

return i;

}

return -1;

}

public static void main(String[] args) throws IOException {

Scanner in = new Scanner(System.in);

int i;

String path = "C:\\java\\employee.txt";

//WRITING

FileOutputStream f = new FileOutputStream(path);

ObjectOutputStream o = new ObjectOutputStream(f);

System.out.print("Enter no:of records: ");

int n = in.nextInt();

Employee[] E = new Employee[n];

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

E[i]=new Employee();

o.writeObject(E[i]);

System.out.println("Record has been added!");

}

//READING

FileInputStream fi = new FileInputStream(path);

ObjectInputStream oi = new ObjectInputStream(fi);

Employee [] E1 = new Employee[n];

System.out.println("\nID\tName\tDept\tDesig");

for (i = 0; i < n+2; i++) {

try{

E1[i] = (Employee) oi.readObject();

System.out.println(E1[i]);

}catch(Exception e){}

}

int ch, ID, No;

String DEPT;

while(true){

System.out.println("\n1.Display Details 2.Department wise Employees 3.Change Desig 4.Change Dept 5.Exit");

System.out.print("Enter your choice: ");

ch=in.nextInt();

if(ch==1)

{

System.out.print("\nEnter Employee ID: ");

ID=in.nextInt();

No=eno(ID, E1, n);

if(No!=-1)

{

System.out.println("\nID\tName\tDept\tDesig");

System.out.println(E1[No]);

}

else

System.out.println("Invalid Employee ID!");

}

else if(ch==2)

{

System.out.print("\nEnter Department: ");

DEPT=in.next();

System.out.println("\nID\tName");

for(i=0;i<n;i++)

{

if(E1[i].dept.equals(DEPT))

System.out.println(E1[i].id +"\t"+ E1[i].name);

}

}

else if(ch==3)

{

System.out.print("\nEnter Employee ID: ");

ID=in.nextInt();

No=eno(ID, E1, n);

if(No!=-1)

{

System.out.print("Enter new Desig: ");

E1[No].desig=in.next();

for (i = 0; i < n; i++)

o.writeObject(E1[i]);

}

else

System.out.println("Invalid Employee ID!");

}

else if(ch==4)

{

System.out.print("\nEnter Employee ID: ");

ID=in.nextInt();

No=eno(ID, E1, n);

if(No!=-1)

{

System.out.print("Enter new Dept: ");

E1[No].dept=in.next();

for (i = 0; i < n; i++)

o.writeObject(E1[i]);

}

else

System.out.println("Invalid Employee ID!");

}

else if(ch==5)

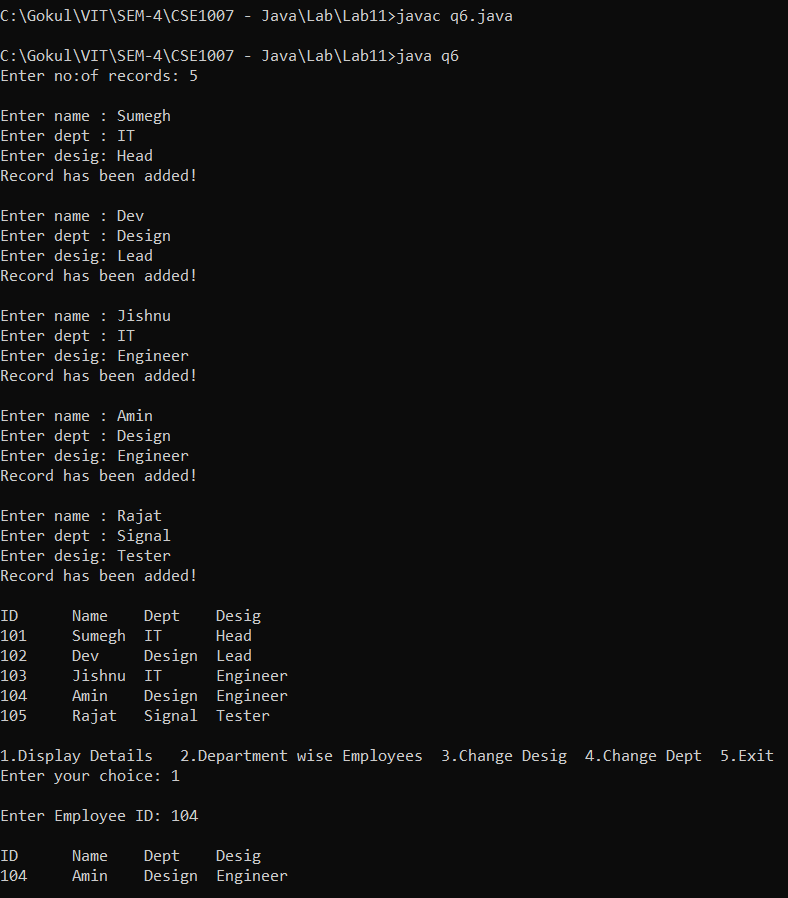
break;

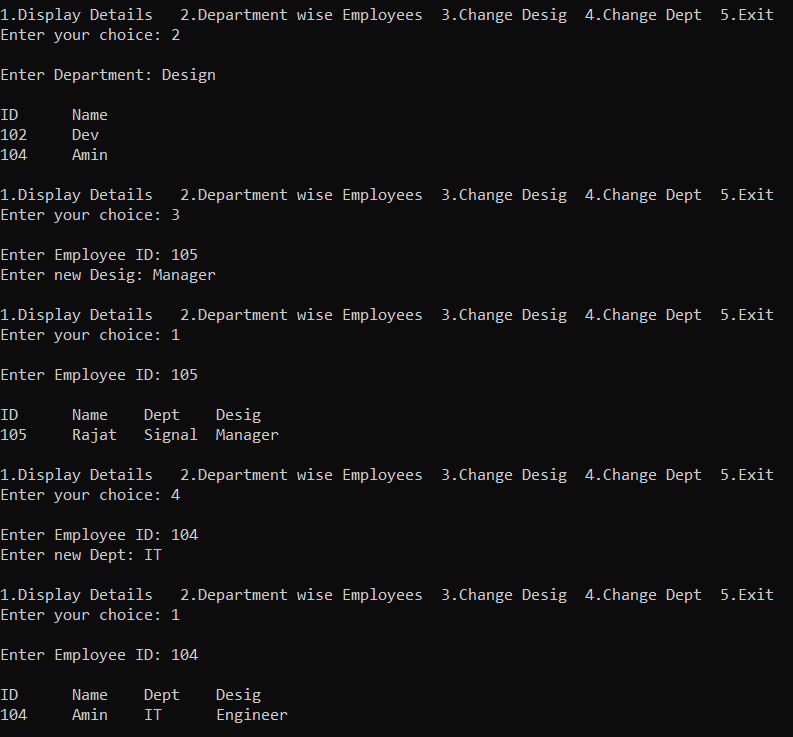
}

}

}

**OUTPUT:**

****

****