# Input Output Statements

## Question 1:

**import** java.util.Scanner;

**public** **class** Main {

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

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

**int** number = input.nextInt();

System.*out*.println("Harry has to swim at " + number +" meters and must rescue "+ number +" sea maidens.");

}

}

## Question 2:

**import** java.util.Scanner;

**public** **class** Main {

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

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

String string = input.next();

System.*out*.println("Harry has to pick the flag that has "+ string +" in it.");

}

}

## Question 3:

**import** java.util.Scanner;

**public** **class** Main {

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

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

Double value = input.nextDouble();

String string = String.*format*("%.2f", value);

System.*out*.println("Harry has got :\nProf.Dumbledore must get the wand worth $"+ string+".");

}

}

## Question 4:

**import** java.util.Scanner;

**public** **class** Main {

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

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

String name = input.next();

System.*out*.println("Fang says :\nHagrid must choose the cracker with name \""+name+"\".");

}

}

## Question 5:

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Word in the Trophy :");

String name1 = input.next();

System.*out*.println("Malfoy picks the character :");

String name2 = input.next();

System.*out*.println("Malfoy replaces the character with :");

String name3 = input.next();

System.*out*.println("Hedwig must replace "+name3+" with "+name2+" in the word "+name1+".");

}

}

# Operators and Expressions

## Question 1: Profit

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the number of dozens of toys purchased");

**int** x = input.nextInt();

System.*out*.println("Enter the price per dozen");

**int** y = input.nextInt();

System.*out*.println("Enter the selling price of 1 toy");

**int** z = input.nextInt();

**double** profitPercentage = ((z-(y/12.0)))/(y/12.0)\*100;

String profitPercentageRounded = String.*format*("%.2f", profitPercentage);

System.*out*.println("Sam's profit percentage is "+ profitPercentageRounded + " percent");

}

}

## Question 2: Time and Work

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the value of x");

**double** x = input.nextInt();

**double** a;

**double** b;

**double** total;

b = (3/2.0)\*x;

a = (1/3.0)\*b;

total = 1/((1/a)+(1/b));

String totalFormat = String.*format*("%.2f", total);

System.*out*.println("Working together, A and B can complete the work in "+totalFormat+" days");

}

}

## Question 3 : Average

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the number of overs bowled so far");

**int** x = input.nextInt();

System.*out*.println("Enter the current run rate");

**double** y = input.nextDouble();

System.*out*.println("Enter the target score");

**int** z = input.nextInt();

**double** runRate;

runRate = (z - (y\*x))/(50-x);

String format = String.*format*("%.2f", runRate);

System.*out*.println("Required run rate is "+format);

}

}

## Question 4: Radius of circle

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the sides of a triangle");

**int** a=input.nextInt();

**int** b=input.nextInt();

**int** c=input.nextInt();

**double** k = (a+b+c)/2.0;

**double** Radius = (**float**) (Math.*sqrt*(k\*(k-a)\*(k-b)\*(k-c)))/k;

String format = String.*format*("%.2f", Radius);

System.*out*.println("The radius of the circle is "+format);

}

}

## Question 5: Pogo Jump Stick

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the X and Y coordinate of friend's house");

**int** x=input.nextInt();

**int** y=input.nextInt();

**double** temp = (x-3)\*(x-3)+(y-4)\*(y-4);

**double** distance = Math.*sqrt*(temp);

**int** format = (**int**) Math.*ceil*(distance);

System.*out*.println("Raju needs "+format+" jumps");

}

}

## Question 6: Discount

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Price of item 1 :");

**double** price1 = in.nextDouble();

System.*out*.println("Price of item 2 :");

**double** price2 = in.nextDouble();

System.*out*.println("Discount in percentage :");

**double** discountPercent = in.nextDouble();

**double** totalPrice = price1 + price2;

**double** savings = totalPrice \* (discountPercent / 100);

**double** discountedAmount = totalPrice - savings;

String disStr = String.*format*("%.2f", discountedAmount);

String totStr = String.*format*("%.2f", totalPrice);

String savStr = String.*format*("%.2f", savings);

System.*out*.println("Total amount : $" + totStr);

System.*out*.println("Discounted amount : $" + disStr);

System.*out*.println("Saved amount : $" + savStr);

}

}

# Conditional Statements

## Question 1: Compare 2 integers

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the first number");

**int** number1 = input.nextInt();

System.*out*.println("Enter the second number");

**int** number2 = input.nextInt();

**if** (number1 > number2) {

System.*out*.println(number1 + " is greater than " + number2);

} **else** **if** (number1 < number2) {

System.*out*.println(number1 + " is less than " + number2);

} **else** {

System.*out*.println(number1 + " is equal to " + number2);

}

}

}

## Question 2: vowel or consonant

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter a character");

**char** character = input.next().charAt(0);

**if** ((character >= 'a' && character <= 'z')

|| (character >= 'A' && character <= 'Z')) {

**switch** (character) {

**case** 'A':

**case** 'a':

**case** 'E':

**case** 'e':

**case** 'I':

**case** 'i':

**case** 'O':

**case** 'o':

**case** 'U':

**case** 'u':

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

**break**;

**default**:

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

}

} **else** {

System.*out*.println("Not an alphabet");

}

}

}

## Question 3: Grade

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the marks");

**int** mark = input.nextInt();

**if**(mark>100 || mark < 0 ){

System.*out*.println("Invalid Input");

}

**else**{

**if**(mark==100){

System.*out*.println("The student obtained a S grade");

}

**else** **if**(mark>=90){

System.*out*.println("The student obtained a A grade");

}

**else** **if**(mark>=80){

System.*out*.println("The student obtained a B grade");

}

**else** **if**(mark>=70){

System.*out*.println("The student obtained a C grade");

}

**else** **if**(mark>=60){

System.*out*.println("The student obtained a D grade");

}

**else** **if**(mark>=50){

System.*out*.println("The student obtained a E grade");

}

**else**{

System.*out*.println("The student obtained a F grade");

}

}

}

}

## Question 4: Profit or Loss

**import** java.util.Scanner;

**public** **class** Main {

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

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

**float** x;

**float** y;

**float** profitLoss;

System.*out*.println("Enter the price of a dozen mangoes");

x = input.nextFloat();

System.*out*.println("Enter the price at which 1 mango is being sold");

y = input.nextFloat();

profitLoss = 12 \* y - x;

**if** (profitLoss == 0) {

System.*out*.println("No profit nor loss");

} **else** **if** (profitLoss > 0) {

System.*out*.println("Profit : Rs."

+ String.*format*("%.2f", profitLoss));

} **else** {

System.*out*.println("Loss : Rs."

+ String.*format*("%.2f", (-profitLoss)));

}

}

}

## Question 5: Fee Collection

**import** java.util.Scanner;

**public** **class** Main {

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

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

String studentType;

**float** tutionFee;

**float** busFee;

**float** hostelFee;

**float** total = 0;

System.*out*.println("Enter the student type");

studentType = input.next();

System.*out*.println("Enter tuition fee");

tutionFee = input.nextFloat();

System.*out*.println("Enter bus fee");

busFee = input.nextFloat();

System.*out*.println("Enter hostel fee");

hostelFee = input.nextFloat();

**if** (studentType.equals("MSDS")) {

total = tutionFee + busFee;

} **else** **if** (studentType.equals("MSH")) {

total = tutionFee + hostelFee;

} **else** **if** (studentType.equals("MGSDS")) {

total = tutionFee \* 1.5f + busFee;

} **else** **if** (studentType.equals("MGSH")) {

total = tutionFee \* 1.5f + hostelFee;

} **else** {

}

String form = String.*format*("%.2f", total);

System.*out*.println("The fees to be paid by the student is Rs."

+ form);

}

}

## Question 6: Lab Allocation I

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter x");

**int** x = input.nextInt();

System.*out*.println("Enter y");

**int** y = input.nextInt();

System.*out*.println("Enter z");

**int** z = input.nextInt();

**if**(x <y && x < z){

System.*out*.println("L1 has the minimal seating capacity");

}

**else** **if**(y < x && y < z){

System.*out*.println("L2 has the minimal seating capacity");

}

**else** **if**(z <x && z < y){

System.*out*.println("L3 has the minimal seating capacity");

}

}

}

## Question 7: Lab Allocation II

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter x");

**int** x = input.nextInt();

System.*out*.println("Enter y");

**int** y = input.nextInt();

System.*out*.println("Enter z");

**int** z = input.nextInt();

System.*out*.println("Enter the lab allocated for ACE training");

String training = input.next();

**if** (training.equals("L1")) {

**if** (y < z) {

System.*out*.println("L2 has the minimal seating capacity");

} **else** {

System.*out*.println("L3 has the minimal seating capacity");

}

} **else** **if** (training.equals("L2")) {

**if** (x < z) {

System.*out*.println("L1 has the minimal seating capacity");

} **else** {

System.*out*.println("L3 has the minimal seating capacity");

}

} **else** **if** (training.equals("L3")) {

**if** (x < y) {

System.*out*.println("L1 has the minimal seating capacity");

} **else** {

System.*out*.println("L2 has the minimal seating capacity");

}

}

}

}

## Question 8: Lab Allocation III

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter x");

**int** x = input.nextInt();

System.*out*.println("Enter y");

**int** y = input.nextInt();

System.*out*.println("Enter z");

**int** z = input.nextInt();

System.*out*.println("Enter n");

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

**int** count = 0;

**if** (n <= x) {

count++;

}

**if** (n <= y) {

count++;

}

**if** (n <= z) {

count++;

}

**if** (count > 0) {

System.*out*.println(count + " lab(s) can accommodate " + n

+ " students");

} **else** {

System.*out*.println("None of the labs can accommodate " + n

+ " students");

}

}

}

## Question 9: Lab Allocation IV

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter x");

**int** x = input.nextInt();

System.*out*.println("Enter y");

**int** y = input.nextInt();

System.*out*.println("Enter z");

**int** z = input.nextInt();

System.*out*.println("Enter n");

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

**if** (n > x && n > y && n > z) {

System.*out*.println("None of the labs can be allocated");

}

**else** {

**int** x1 = (x - n >= 0) ? x : Integer.*MAX\_VALUE*;

**int** y1 = (y - n >= 0) ? y : Integer.*MAX\_VALUE*;

**int** z1 = (z - n >= 0) ? z : Integer.*MAX\_VALUE*;

**if** (x1 < y1 && x1 < z1) {

System.*out*.println("L1 should be allocated to this class");

}

**else** **if** (y1 < z1 && y1 < x1) {

System.*out*.println("L2 should be allocated to this class");

}

**else** **if** (z1 < x1 && z1 < y1) {

System.*out*.println("L3 should be allocated to this class");

}

}

}

}

## Question 10: Seat Allocation I

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the number of rows");

**int** r = input.nextInt();

System.*out*.println("Enter the number of columns");

**int** c = input.nextInt();

System.*out*.println("Enter the roll number of the student");

**int** s = input.nextInt();

**if** (s % r == 1 || s % c == 0 || s % r == 0) {

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

} **else** {

System.*out*.println("no");

}

}

}

## Question 11: Seat Allocation II

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the number of rows");

**int** r = input.nextInt();

System.*out*.println("Enter the number of columns");

**int** c = input.nextInt();

System.*out*.println("Enter the roll number of the student");

**int** s = input.nextInt();

**if** (s % r == 1 || s <= r || s >= ((r \* c) - r)) {

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

} **else** {

System.*out*.println("no");

}

}

}

## Question 12: P2 Green Lights

**import** java.util.Scanner;

**public** **class** Main {

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

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

**float** speed = input.nextFloat();

**float** distance = input.nextFloat();

**float** time = input.nextFloat();

**if** ((time/10.0) < ((distance / speed)\*360)) {

System.*out*.println("no");

} **else** {

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

}

}

}

## Question 13: P1 Dining

**import** java.util.Scanner;

**public** **class** Main {

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

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

String side = input.next();

**int** rail = input.nextInt();

**if** (side.equals("front")) {

**if** (rail == 1) {

System.*out*.println("Left Handed");

} **else** **if** (rail == 2) {

System.*out*.println("Right Handed");

}

} **else** **if** (side.equals("rear")) {

**if** (rail == 2) {

System.*out*.println("Left Handed");

} **else** **if** (rail == 1) {

System.*out*.println("Right Handed");

}

}

}

}

## Question 14: Microwave oven

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the number of items");

**int** item = input.nextInt();

System.*out*.println("Enter the single item heating time");

**float** time = input.nextFloat();

**if** (item == 1) {

System.*out*.println("The recommended heating time is "

+ String.*format*("%.2f", time));

} **else** **if** (item == 2) {

System.*out*.println("The recommended heating time is "

+ String.*format*("%.2f", time \* 1.5));

} **else** **if** (item == 3) {

System.*out*.println("The recommended heating time is "

+ String.*format*("%.2f", time \* 2));

} **else** **if** (item > 3) {

System.*out*.println("Number of items is more ");

}

}

}

## Question 15: Matinee movie ticket

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter your age");

**int** age = in.nextInt();

System.*out*.println("Enter show timing");

**double** showTime = in.nextDouble();

**double** price = 0;

**if** (age > 13) {

**if** (showTime == 13.30) {

price = 5;

} **else** {

price = 8;

}

} **else** **if** (age <= 13 && age > 0) {

**if** (showTime == 13.30) {

price = 2;

} **else** {

price = 4;

}

}

String str = String.*format*("%.2f", price);

System.*out*.println("The ticket price is $" + str);

}

}

# Looping Statements

## Question 1: Factorial Number

**import** java.util.Scanner;

**public** **class** Main {

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

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

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

**int** flag = 0;

**for** (**int** i = 1; flag == 0 && n != 1; i++) {

**if** (n % i == 0) {

n = n / i;

} **else** {

flag = 1;

}

}

**if** (flag == 1) {

System.*out*.println("no");

} **else** {

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

}

}

}

## \* Question 2: Lucus sequence

**import** java.util.Scanner;

**public** **class** Main {

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

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

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

**int** a = -1;

**int** b = 1;

**int** c = 1;

**int** temp;

System.*out*.print(a);

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

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

**for** (**int** i = 0; i < n - 3; i++) {

temp = a + b + c;

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

a = b;

b = c;

c = temp;

}

}

}

## Question 3: Fibonacci

**import** java.util.Scanner;

**public** **class** Main {

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

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

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

**int** a = -1;

**int** b = 1;

**int** temp = 0;

**int** flag = 0;

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

temp = a + b;

a = b;

b = temp;

**if** (n == temp) {

flag = 1;

} **else** {

flag = 0;

}

}

String str = (flag == 1) ? "yes" : "no";

System.*out*.println(str);

}

}

## Question 4: Trendy numbers

**import** java.util.Scanner;

**public** **class** Main {

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

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

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

**int** num = n / 10;

num %= 10;

**if** (n >= 100 && n <= 999 && num % 3 == 0) {

System.*out*.println("Trendy Number");

} **else** {

System.*out*.println("Not a Trendy Number");

}

}

}

## Question 5: Special Number

**import** java.util.Scanner;

**public** **class** Main {

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

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

**int** m = input.nextInt();

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

**int** i;

**int** j;

**for** (**int** k = m; k <= n; k++) {

i = k / 10;

j = k % 10;

**if** ((i \* j) + (i + j) == k) {

System.*out*.print(k);

}

}

}

}

## Question 6 : Kepraker Number

**import** java.util.Scanner;

**public** **class** Main {

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

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

**int** m = input.nextInt();

Long k = (**long**) (m \* m);

String num = k.toString();

String number1 = (num.substring(0, (num.length() / 2)));

String number2 = (num.substring(num.length() / 2));

**if** (number1.length() == 0) {

number1 = "0";

}

**long** temp = (Long.*parseLong*(number1) + Long.*parseLong*(number2));

**if** (temp == m) {

System.*out*.println("Kaprekar Number");

} **else** {

System.*out*.println("Not A Kaprekar Number");

}

}

}

## Question 7: Target practice

**import** java.util.Scanner;

**public** **class** Main {

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

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

**int** total = input.nextInt();

**int** count = 0;

**int** temp = 0;

**int** sum = 0;

**while** (sum < total) {

temp = input.nextInt();

sum += temp;

count++;

}

System.*out*.println("The number of turns is " + count);

}

}

## Question 8: Kid’s Homework

**import** java.util.Scanner;

**public** **class** Main {

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

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

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

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

**for** (**int** j = 1; j <= i; j++) {

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

}

System.*out*.println();

}

}

}

# Arrays and Strings

## Question 1: Modulo

**import** java.util.Scanner;

**public** **class** Main {

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

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

**int** temp;

**int** cnt = 0;

**int** flag = 0;

**int** count[] = **new** **int**[10];

**for** (**int** i = 0; i < 10; i++) {

flag = 0;

temp = input.nextInt();

**for** (**int** j = 0; j < cnt; j++) {

**if** (count[j] == temp % 42) {

flag = 1;

}

}

**if** (flag == 0) {

count[i] = temp % 42;

cnt++;

}

}

System.*out*.println(cnt);

}

}

## Question 2: Carry operation

**import** java.util.Scanner;

**public** **class** Main {

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

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

**long** num1 = in.nextLong();

**long** num2 = in.nextLong();

**long** temp;

**long** carry = 0;

**long** count = 0;

**while** (num1 != 0 || num2 != 0) {

temp = (num1 % 10) + (num2 % 10) + carry;

**if** (temp >= 10) {

carry = 1;

count++;

} **else** {

carry = 0;

}

num1 = num1 / 10;

num2 = num2 / 10;

}

System.*out*.println(count);

}

}

## Question 3: Statues

**import** java.util.Scanner;

**public** **class** Main {

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

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

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

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

**int** count = 0;

**int** sum = 0;

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

room[i] = input.nextInt();

sum += room[i];

}

sum /= n;

**for** (**int** i = 0; i < room.length; i++) {

**if** (room[i] - sum > 0) {

count += (room[i] - sum);

}

}

System.*out*.println(count);

}

}

## Question 4: Magic number

**import** java.util.Scanner;

**public** **class** Main {

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

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

**int** num = in.nextInt();

**int** temp = num;

String str = Integer.*toString*(num);

**int** flag = 0;

**int** i = 1;

**int** count = 0;

**while** (flag == 0 && temp != 0) {

**if** (temp % 10 == 4) {

num = num + (3 \* i);

flag = 1;

} **else** **if** (temp % 10 == 7) {

num = num - (3 \* i);

i \*= 10;

temp = temp / 10;

}

count++;

}

**if** (count == str.length() && flag == 0) {

num = num + 4 \* i;

}

System.*out*.println(num);

}

}

## Question 5: Chess Puzzle

**import** java.util.Scanner;

**public** **class** Main {

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

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

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

**int** order[][] = **new** **int**[2][n];

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

order[0][i] = input.nextInt();

order[1][i] = input.nextInt();

}

**boolean** flag = **false**;

**for** (**int** i = 0; i < n - 1 && flag == **false**; i++) {

**for** (**int** k = i + 1; k < n && flag == **false**; k++) {

**if** ((order[0][i] == order[0][k])

|| (order[1][i] == order[1][k])) {

flag = **true**;

}

}

}

**if** (flag == **true**) {

System.*out*.println("NOT SAFE");

} **else** {

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

}

}

}

## Question 6: Cheer Leader Pattern

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the size of matrix:");

**int** num = in.nextInt();

// char matrix[][]= new char[n][n];

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

**for** (**int** j = 0; j < num; j++) {

**if** (i == j || i + j == num - 1) {

System.*out*.print("M ");

} **else** **if** (i == 0 || j == 0 || i == num - 1 || j == num - 1) {

System.*out*.print("F ");

} **else** {

System.*out*.print("C ");

}

}

System.*out*.println();

}

}

}

## Question 7: Counting Stars

**import** java.util.Scanner;

**public** **class** Main {

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

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

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

**int** m = input.nextInt();

**int** data[][] = **new** **int**[n][m];

**int** sum = 0;

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

**for** (**int** j = 0; j < m; j++) {

data[i][j] = input.nextInt();

}

}

**int** noOfQueries = input.nextInt();

**int** query[][] = **new** **int**[noOfQueries][4];

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

**for** (**int** j = 0; j < 4; j++) {

query[i][j] = input.nextInt();

}

}

**for** (**int** k = 0; k < noOfQueries; k++) {

sum = 0;

**for** (**int** i = query[k][0] - 1; i < query[k][2]; i++) {

**for** (**int** j = query[k][1] - 1; j < query[k][3]; j++) {

sum += data[i][j];

}

}

System.*out*.println(sum);

}

}

}

## Question 8: String Encryption

**import** java.util.Scanner;

**public** **class** Main {

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

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

String string = input.next();

**char** data[] = **new** **char**[string.length()];

**int** number[] = **new** **int**[string.length()];

**int** count = 0;

**boolean** flag = **false**;

**for** (**int** i = 0; i < string.length(); i++) {

flag = **false**;

**for** (**int** j = 0; j < count && flag == **false**; j++) {

**if** (data[j] == string.charAt(i)) {

number[j]++;

flag = **true**;

}

}

**if** (flag == **false**) {

data[count] = string.charAt(i);

number[count]++;

count++;

}

}

String str = "";

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

str += data[i];

str += number[i];

}

System.*out*.println(str);

}

### }

## Question 9: Distorted Message

**import** java.util.Scanner;

**public** **class** Main {

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

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

String string = input.nextLine();

string = string.replaceAll("iiing", "th");

System.*out*.println(string);

}

### }

## Question 10: YODA Talk

**import** java.util.Scanner;

**public** **class** Main {

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

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

String string = input.nextLine();

String str[] = string.split(" ");

string = "";

**for** (**int** i = 2; i < str.length; i++) {

string += str[i] + " ";

}

string += str[0] + " " + str[1];

System.*out*.println(string);

}

### }

# String API

## Question 1: lastIndexOf()

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the string");

String string = input.nextLine();

System.*out*.println("Enter the string to be searched");

String str = input.next();

System.*out*.println("The index of last occurence of \"" + str

+ "\" is " + string.lastIndexOf(str));

System.*out*.println("Enter the index limit");

**int** index = input.nextInt();

System.*out*.println("First occurence of \"" + str + "\" from " + index

+ "th index backwards is " + string.lastIndexOf(str, index));

}

### }

## Question 2: startsWith()

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the string");

String string = input.nextLine();

System.*out*.println("Enter the start string");

String str = input.nextLine();

**if** (string.startsWith(str)) {

System.*out*

.println("\"" + string + "\" starts with \"" + str + "\"");

} **else** {

System.*out*.println("\"" + string + "\" does not start with \""

+ str + "\"");

}

}

}

## Question 3: endsWith()

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the string");

String string = input.next();

System.*out*.println("Enter the end string");

String str = input.next();

**if** (string.endsWith(str)) {

System.*out*.println("\"" + string + "\" ends with \"" + str + "\"");

} **else** {

System.*out*.println("\"" + string + "\" does not end with \"" + str

+ "\"");

}

}

## }

## Question 4: split()

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the string");

String string = input.nextLine();

System.*out*.println("The words in the string are");

String str[] = string.split(" ");

**for** (**int** i = 0; i < str.length; i++) {

**if** (str[i].length() != 0) {

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

}

}

}

}

## Question 5:replaceAll()

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the content of the document");

String content = input.nextLine();

System.*out*.println("Enter the old name of the company");

String oldName = input.next();

System.*out*.println("Enter the new name of the company");

String newName = input.next();

content = content.replaceAll(oldName, newName);

System.*out*.println("The content of the modified document is");

System.*out*.println(content);

}

}

## Question 5: Removing multiple spaces

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the string");

String string = input.nextLine();

System.*out*.print("The processed string is");

String str[] = string.split(" ");

**for** (**int** i = 0; i < str.length; i++) {

**if** (str[i].length() != 0) {

System.*out*.print(" " + str[i]);

}

}

}

}

## Question 6: Display String Vertically

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the string");

String string = input.nextLine();

System.*out*

.println("The string printed vertically forwards and backwards is");

String str[] = string.split("");

**for** (**int** i = 0; i < str.length; i++) {

**if** (str[i].length() != 0) {

System.*out*.println(str[i] +" "+ str[str.length - i]);

}

}

}

}

## Question 8 : Adjecent characters

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the string");

String string = input.nextLine();

**char** str[] = string.toCharArray();

String stringNew = "" + str[0];

**for** (**int** i = 1; i < string.length(); i++) {

**if** (str[i - 1] == str[i]) {

stringNew += '\*';

stringNew += str[i];

} **else** {

stringNew += str[i];

}

}

System.*out*.println("The processed string is " + stringNew);

}

}

## Question 9: Move Lower x

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the string");

String string = input.nextLine();

String temp = "";

**char** str[] = string.toCharArray();

String stringNew = "";

**for** (**int** i = 0; i < string.length(); i++) {

**if** (str[i] != 'x') {

stringNew += str[i];

} **else** {

temp += str[i];

}

}

System.*out*.println("The processed string is " + stringNew + temp);

}

}

## Question 10 : Capitalize String

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the string");

String string = input.nextLine();

String stringNew = "";

String str[] = string.split(" ");

**for** (**int** i = 0; i < str.length; i++) {

str[i] = str[i].substring(0, 1).toUpperCase() + str[i].substring(1);

stringNew += str[i] + " ";

}

System.*out*.println("Capitalized version:");

System.*out*.println(stringNew);

}

}

# Object Oriented Programming I

## Question 1: Getters and Setters

**import** java.util.Scanner;

**class** Product {

**private** Long id;

**private** String productName;

**private** String supplierName;

**public** Long getId() {

**return** id;

}

**public** **void** setId(Long id) {

**this**.id = id;

}

**public** String getProductName() {

**return** productName;

}

**public** **void** setProductName(String productName) {

**this**.productName = productName;

}

**public** String getSupplierName() {

**return** supplierName;

}

**public** **void** setSupplierName(String supplierName) {

**this**.supplierName = supplierName;

}

}

**public** **class** Main {

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

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

Product object = **new** Product();

System.*out*.println("Enter the product id");

object.setId(input.nextLong());

System.*out*.println("Enter the product name");

object.setProductName(input.next());

System.*out*.println("Enter the supplier name");

object.setSupplierName(input.next());

System.*out*.println("Product Id is " + object.getId());

System.*out*.println("Product Name is " + object.getProductName());

System.*out*.println("Supplier Name is " + object.getSupplierName());

}

}

## Question 2: Constuctors

**import** java.util.Scanner;

**public** **class** Main {

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

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

Product object = **new** Product();

System.*out*.println("Enter the product id");

object.setId(input.nextLong());

input.nextLine();

System.*out*.println("Enter the product name");

object.setProductName(input.nextLine());

System.*out*.println("Enter the supplier name");

object.setSupplierName(input.nextLine());

System.*out*.println("Product Id is " + object.getId());

System.*out*.println("Product Name is " + object.getProductName());

System.*out*.println("Supplier Name is " + object.getSupplierName());

}

}

**public** **class** Product {

**private** Long id;

**private** String productName;

**private** String supplierName;

**public** Product() {

}

**public** Product(Long id, String productName, String supplierName) {

**this**.id = id;

**this**.productName = productName;

**this**.supplierName = supplierName;

}

**public** Long getId() {

**return** id;

}

**public** **void** setId(Long id) {

**this**.id = id;

}

**public** String getProductName() {

**return** productName;

}

**public** **void** setProductName(String productName) {

**this**.productName = productName;

}

**public** String getSupplierName() {

**return** supplierName;

}

**public** **void** setSupplierName(String supplierName) {

**this**.supplierName = supplierName;

}

**void** display() {

System.*out*.println("Product Id is " + **this**.id);

System.*out*.println("Product Name is " + **this**.productName);

System.*out*.println("Supplier Name is " + **this**.supplierName);

}

**public** String toString() {

**return** (**this**.id + " : " + **this**.productName + " : " + **this**.supplierName);

}

**public** **boolean** equals(Product temp) {

**boolean** flag;

**if** (**this**.id == temp.id && **this**.productName.equals(temp.productName)

&& **this**.supplierName.equals(temp.supplierName)) {

flag = **true**;

} **else** {

flag = **false**;

}

**return** flag;

}

}

## Question 3: Constructor Overloading

**import** java.util.Scanner;

**public** **class** Main {

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

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

Product object = **new** Product();

System.*out*.println("Enter the product id");

object.setId(input.nextLong());

input.nextLine();

System.*out*.println("Enter the product name");

object.setProductName(input.nextLine());

System.*out*

.println("Is the product supplied by Nivas Suppliers? Type yes or no (not case sensitive)");

**if** (input.next().equalsIgnoreCase("yes")) {

object.setSupplierName("Nivas");

} **else** {

input.nextLine();

System.*out*.println("Enter the supplier name");

object.setSupplierName(input.nextLine());

}

System.*out*.println("Product Id is " + object.getId());

System.*out*.println("Product Name is " + object.getProductName());

System.*out*.println("Supplier Name is " + object.getSupplierName());

}

}

**public** **class** Product {

**private** Long id;

**private** String productName;

**private** String supplierName;

**public** Product() {

}

**public** Product(Long id, String productName) {

**this**.id = id;

**this**.productName = productName;

**this**.supplierName = "Nivas";

}

**public** Product(Long id, String productName, String supplierName) {

**this**.id = id;

**this**.productName = productName;

**this**.supplierName = supplierName;

}

**public** Long getId() {

**return** id;

}

**public** **void** setId(Long id) {

**this**.id = id;

}

**public** String getProductName() {

**return** productName;

}

**public** **void** setProductName(String productName) {

**this**.productName = productName;

}

**public** String getSupplierName() {

**return** supplierName;

}

**public** **void** setSupplierName(String supplierName) {

**this**.supplierName = supplierName;

}

**void** display() {

System.*out*.println("Product Id is " + **this**.id);

System.*out*.println("Product Name is " + **this**.productName);

System.*out*.println("Supplier Name is " + **this**.supplierName);

}

**public** String toString() {

**return** (**this**.id + " : " + **this**.productName + " : " + **this**.supplierName);

}

**public** **boolean** equals(Product temp) {

**boolean** flag;

**if** (**this**.id == temp.id && **this**.productName.equals(temp.productName)

&& **this**.supplierName.equals(temp.supplierName)) {

flag = **true**;

} **else** {

flag = **false**;

}

**return** flag;

}

}

## Question 4: toString() and getClass()

**import** java.util.Scanner;

**public** **class** Main {

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

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

Product object = **new** Product();

System.*out*.println("Enter the product id");

object.setId(input.nextLong());

input.nextLine();

System.*out*.println("Enter the product name");

object.setProductName(input.nextLine());

System.*out*.println("Enter the supplier name");

object.setSupplierName(input.nextLine());

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

System.*out*.print("Invoking getClass() method : " + object.getClass());

}

}

**public** **class** Product {

**private** Long id;

**private** String productName;

**private** String supplierName;

**public** Product() {

}

**public** Product(Long id, String productName) {

**this**.id = id;

**this**.productName = productName;

**this**.supplierName = "Nivas";

}

**public** Product(Long id, String productName, String supplierName) {

**this**.id = id;

**this**.productName = productName;

**this**.supplierName = supplierName;

}

**public** Long getId() {

**return** id;

}

**public** **void** setId(Long id) {

**this**.id = id;

}

**public** String getProductName() {

**return** productName;

}

**public** **void** setProductName(String productName) {

**this**.productName = productName;

}

**public** String getSupplierName() {

**return** supplierName;

}

**public** **void** setSupplierName(String supplierName) {

**this**.supplierName = supplierName;

}

**void** display() {

System.*out*.println("Product Id is " + **this**.id);

System.*out*.println("Product Name is " + **this**.productName);

System.*out*.println("Supplier Name is " + **this**.supplierName);

}

**public** String toString() {

**return** (**this**.id + " : " + **this**.productName + " : " + **this**.supplierName);

}

**public** **boolean** equals(Product temp) {

**boolean** flag;

**if** (**this**.id == temp.id && **this**.productName.equals(temp.productName)

&& **this**.supplierName.equals(temp.supplierName)) {

flag = **true**;

} **else** {

flag = **false**;

}

**return** flag;

}

}

## Question 5: equals()

**import** java.util.Scanner;

**public** **class** Main {

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

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

Product obj1 = **new** Product();

System.*out*.println("Enter the product id");

obj1.setId(input.nextLong());

System.*out*.println("Enter the product name");

obj1.setProductName(input.next());

System.*out*.println("Enter the supplier name");

obj1.setSupplierName(input.next());

obj1.display();

Product obj2 = **new** Product();

System.*out*.println("Enter the product id");

obj2.setId(input.nextLong());

System.*out*.println("Enter the product name");

obj2.setProductName(input.next());

System.*out*.println("Enter the supplier name");

obj2.setSupplierName(input.next());

obj2.display();

**if** (obj1.equals(obj2)) {

System.*out*.println("The two products are the same");

} **else** {

System.*out*.println("The two products are different");

}

}

}

**public** **class** Product {

**private** Long id;

**private** String productName;

**private** String supplierName;

**public** Product() {

}

**public** Product(Long id, String productName, String supplierName) {

**this**.id = id;

**this**.productName = productName;

**this**.supplierName = supplierName;

}

**public** Long getId() {

**return** id;

}

**public** **void** setId(Long id) {

**this**.id = id;

}

**public** String getProductName() {

**return** productName;

}

**public** **void** setProductName(String productName) {

**this**.productName = productName;

}

**public** String getSupplierName() {

**return** supplierName;

}

**public** **void** setSupplierName(String supplierName) {

**this**.supplierName = supplierName;

}

**void** display() {

System.*out*.println("Product Id is " + **this**.id);

System.*out*.println("Product Name is " + **this**.productName);

System.*out*.println("Supplier Name is " + **this**.supplierName);

}

**public** String toString() {

**return** (**this**.id + " : " + **this**.productName + " : " + **this**.supplierName);

}

**public** **boolean** equals(Product temp) {

**boolean** flag;

**if** (**this**.id == temp.id && **this**.productName.equals(temp.productName)

&& **this**.supplierName.equals(temp.supplierName)) {

flag = **true**;

} **else** {

flag = **false**;

}

**return** flag;

}

}

## Question 6: Static members

**import** java.util.Scanner;

**public** **class** Main {

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

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

StaticIllustration obj1 = **new** StaticIllustration();

System.*out*.println("Enter Object 1 details");

System.*out*.println("Enter i1");

obj1.setI1(input.nextInt());

System.*out*.println("Enter i2");

StaticIllustration.*setI2*(input.nextInt());

System.*out*.print("Object 1 Details : ");

System.*out*.println(obj1.toString());

StaticIllustration obj2 = **new** StaticIllustration();

System.*out*.println("Enter Object 2 details");

System.*out*.println("Enter i1");

obj2.setI1(input.nextInt());

System.*out*.println("Enter i2");

StaticIllustration.*setI2*(input.nextInt());

System.*out*.print("Object 2 Details : ");

System.*out*.println(obj2.toString());

System.*out*.print("Object 1 Details : ");

System.*out*.println(obj1.toString());

}

}

**public** **class** StaticIllustration {

**public** **static** **int** *count*;

**public** **int** no;

**private** Integer i1;

**private** **static** Integer *i2*;

**public** StaticIllustration() {

*count*++;

**this**.no = *count*;

}

**public** Integer getI1() {

**return** i1;

}

**public** **void** setI1(Integer i1) {

**this**.i1 = i1;

}

**public** **static** Integer getI2() {

**return** *i2*;

}

**public** **static** **void** setI2(Integer i2) {

StaticIllustration.*i2* = i2;

}

@Override

**public** String toString() {

**return** "i1 = " + **this**.i1 + ",i2 = " + StaticIllustration.*i2*;

}

}

## Question 7: No. of objects

**import** java.util.Scanner;

**import** E\_box.Dummy;

**public** **class** Main {

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

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

**int** number;

String reply;

**do** {

Dummy obj = **new** Dummy();

System.*out*.println("Enter Object " + Dummy.getCount()

+ " \"a\" value");

number = input.nextInt();

obj.setA(number);

System.out.println("Object " + Dummy.getCount() + " details");

obj.display();

System.*out*.println("Do you want to create another object?"

+ " Type yes or no (not case sensitive)");

reply = input.next();

} **while** (reply.equalsIgnoreCase("yes"));

System.*out*.println("The number of objects created so far is "

+ Dummy.getCount());

}

}

**package** E\_box;

**public** **class** Dummy {

**private** **int** a;

**private** **static** **int** *count*;

**public** **int** getA() {

**return** a;

}

**public** **void** setA(**int** a) {

**this**.a = a;

}

**public** **static** **int** getCount() {

**return** *count*;

}

**public** **static** **void** setCount(**int** count) {

Dummy.*count* = count;

}

**public** Dummy() {

*count*++;

}

**public** Dummy(**int** a) {

**this**.a = a;

*count*++;

}

**public** **void** display() {

System.*out*.println("The value of a is " + **this**.a);

System.*out*.println("The number of objects created is " + *count*);

}

}

## Question 8: Math Class

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the first integer");

Integer n1 = input.nextInt();

System.*out*.println("Enter the second integer");

Integer n2 = input.nextInt();

System.*out*.println("Absolute value of " + n1 + " is " + Math.*abs*(n1));

System.*out*.println("Absolute value of " + n2 + " is " + Math.*abs*(n2));

**if** (n1.equals(n2)) {

System.*out*.println(n1 + " = " + n2);

} **else** {

System.*out*.println(n1 + " != " + n2);

}

}

}

## Question 9: Wapper Class – Integer I

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter an integer");

Integer number = input.nextInt();

System.*out*.println("The binary equivalent of " + number + " is "

+ Integer.*toBinaryString*(number));

System.*out*.println("The hexadecimal equivalent of " + number + " is "

+ Integer.*toHexString*(number));

System.*out*.println("The octal equivalent of " + number + " is "

+ Integer.*toOctalString*(number));

System.*out*.println("Byte value of " + number + " is "

+ number.byteValue());

System.*out*.println("Short value of " + number + " is "

+ number.shortValue());

System.*out*.println("Long value of " + number + " is "

+ number.longValue());

System.*out*.println("Float value of " + number + " is "

+ number.floatValue());

System.*out*.println("Double value of " + number + " is "

+ number.doubleValue());

}

}

## Question 10: Wapper Class – Integer II

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the binary number");

String number1 = input.next();

System.*out*.println("Enter the octal number");

String number2 = input.next();

System.*out*.println("Enter the hexadecimal number");

String number3 = input.next();

System.*out*.println("The integer value of the binary number " + number1

+ " is " + Integer.*parseInt*(number1, 2));

System.*out*.println("The integer value of the octal number " + number2

+ " is " + Integer.*parseInt*(number2, 8));

System.*out*.println("The integer value of the hexadecimal number "

+ number3 + " is " + Integer.*parseInt*(number3, 16));

}

}

## Question 11: Command Line Arguments 1

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("The number of arguments passed is "

+ args.length);

**if** (args.length > 0) {

System.*out*.println("The command line arguments passed are");

**for** (**int** i = 0; i < args.length; i++) {

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

}

}

}

}

## Question 12: Command Line Arguments 2

**import** java.util.Scanner;

**public** **class** Main {

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

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

**if** (args.length == 2) {

**int** sum = Integer.*valueOf*(args[0]) + Integer.*valueOf*(args[1]);

System.*out*.println("The sum of " + args[0] + " and " + args[1]

+ " is " + sum);

} **else** {

System.*out*.println("Invalid Input");

}

}

}

# Object Oriented Programming II

## Question 1: product 1

**import** java.util.Scanner;

**public** **class** Main {

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

Long id;

Integer price;

String pname, sname;

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

System.*out*.println("Enter the product id");

id = input.nextLong();

input.nextLine();

System.*out*.println("Enter the product name");

pname = input.nextLine();

System.*out*.println("Enter the supplier name");

sname = input.nextLine();

System.*out*.println("Enter the product price");

price = input.nextInt();

Product ob1 = **new** Product(id, pname, sname, price);

ProductBO obj = **new** ProductBO();

obj.displayProductDetails(ob1);

}

}

**public** **class** Product {

**private** Long id;

**private** String productName;

**private** String supplierName;

**private** Integer price;

**public** Long getId() {

**return** id;

}

**public** **void** setId(Long id) {

**this**.id = id;

}

**public** String getProductName() {

**return** productName;

}

**public** **void** setProductName(String productName) {

**this**.productName = productName;

}

**public** String getSupplierName() {

**return** supplierName;

}

**public** **void** setSupplierName(String supplierName) {

**this**.supplierName = supplierName;

}

**public** Integer getPrice() {

**return** price;

}

**public** **void** setPrice(Integer price) {

**this**.price = price;

}

**public** Product() {

}

**public** Product(Long id, String productName, String supplierName,

Integer price) {

**this**.id = id;

**this**.productName = productName;

**this**.supplierName = supplierName;

**this**.price = price;

}

}

**public** **class** ProductBO {

**public** **void** displayProductDetails(Product product) {

System.*out*.println("Product Id is " + product.getId());

System.*out*.println("Product Name is " + product.getProductName());

System.*out*.println("Supplier Name is " + product.getSupplierName());

System.*out*.println("Product price is " + product.getPrice());

}

}

## Question 2: Product 2

**import** java.util.Scanner;

**public** **class** Main {

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

Long id;

Integer price;

String pname, sname;

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

System.*out*.println("Enter the product id");

id = input.nextLong();

input.nextLine();

System.*out*.println("Enter the product name");

pname = input.nextLine();

System.*out*.println("Enter the supplier name");

sname = input.nextLine();

System.*out*.println("Enter the product price");

price = input.nextInt();

Product ob1 = **new** Product(id, pname, sname, price);

ProductBO obj = **new** ProductBO();

obj.displayProductDetails(ob1);

System.*out*.println("Update Menu :");

System.*out*.println("1) Update product name");

System.*out*.println("2) Update supplier name");

System.*out*.println("3) Update product price");

System.*out*.println("Enter Choice");

**int** ch = input.nextInt();

input.nextLine();

**switch** (ch) {

**case** 1:

System.*out*.println("Enter new product name");

pname = input.nextLine();

obj.updateProductName(ob1, pname);

**break**;

**case** 2:

System.*out*.println("Enter new supplier name");

sname = input.nextLine();

obj.updateSupplierName(ob1, sname);

**break**;

**case** 3:

System.*out*.println("Enter new product price");

price = input.nextInt();

obj.updateProductPrice(ob1, price);

**break**;

}

obj.displayProductDetails(ob1);

}

}

**public** **class** Product {

**private** Long id;

**private** String productName;

**private** String supplierName;

**private** Integer price;

**public** Long getId() {

**return** id;

}

**public** **void** setId(Long id) {

**this**.id = id;

}

**public** String getProductName() {

**return** productName;

}

**public** **void** setProductName(String productName) {

**this**.productName = productName;

}

**public** String getSupplierName() {

**return** supplierName;

}

**public** **void** setSupplierName(String supplierName) {

**this**.supplierName = supplierName;

}

**public** Integer getPrice() {

**return** price;

}

**public** **void** setPrice(Integer price) {

**this**.price = price;

}

**public** Product() {

}

**public** Product(Long id, String productName, String supplierName,

Integer price) {

**this**.id = id;

**this**.productName = productName;

**this**.supplierName = supplierName;

**this**.price = price;

}

}

**public** **class** ProductBO {

**public** **void** displayProductDetails(Product product) {

System.*out*.println("Product Id is " + product.getId());

System.*out*.println("Product Name is " + product.getProductName());

System.*out*.println("Supplier Name is " + product.getSupplierName());

System.*out*.println("Product price is " + product.getPrice());

}

**public** **void** updateProductName(Product product, String pname) {

product.setProductName(pname);

}

**public** **void** updateSupplierName(Product product, String sname) {

product.setSupplierName(sname);

}

**public** **void** updateProductPrice(Product product, Integer p) {

product.setPrice(p);

}

}

## Question 3: Product 3

**import** java.util.Scanner;

**public** **class** Main {

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

**int** i = 0;

Long id;

Integer price;

String pname, sname;

String choice;

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

Product ob[] = **new** Product[10];

ProductBO obj = **new** ProductBO();

**do** {

System.*out*.println("Enter the product id");

id = input.nextLong();

input.nextLine();

System.*out*.println("Enter the product name");

pname = input.nextLine();

System.*out*.println("Enter the supplier name");

sname = input.nextLine();

System.*out*.println("Enter the product price");

price = input.nextInt();

ob[i] = **new** Product(id, pname, sname, price);

System.*out*

.println("Do you want to enter the details of another product? Enter yes or no (not case sensitive)");

choice = input.next();

i++;

} **while** (choice.equalsIgnoreCase("YES"));

obj.displayAllProductDetails(ob, i);

}

}

**public** **class** Product {

**private** Long id;

**private** String productName;

**private** String supplierName;

**private** Integer price;

**public** Long getId() {

**return** id;

}

**public** **void** setId(Long id) {

**this**.id = id;

}

**public** String getProductName() {

**return** productName;

}

**public** **void** setProductName(String productName) {

**this**.productName = productName;

}

**public** String getSupplierName() {

**return** supplierName;

}

**public** **void** setSupplierName(String supplierName) {

**this**.supplierName = supplierName;

}

**public** Integer getPrice() {

**return** price;

}

**public** **void** setPrice(Integer price) {

**this**.price = price;

}

**public** Product() {

}

**public** Product(Long id, String productName, String supplierName,

Integer price) {

**this**.id = id;

**this**.productName = productName;

**this**.supplierName = supplierName;

**this**.price = price;

}

}

**public** **class** ProductBO {

**public** **void** displayProductDetails(Product product) {

System.*out*.format("%-10s %-20s %-20s %-10s\n", product.getId(),

product.getProductName(), product.getSupplierName(),

product.getPrice());

}

**public** **void** displayAllProductDetails(Product productList[], **int** count) {

System.*out*.format("%-10s %-20s %-20s %-10s\n", "Id", "ProductName",

"SupplierName", "Price");

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

displayProductDetails(productList[i]);

}

}

**public** **void** updateProductName(Product product, String pname) {

product.setProductName(pname);

}

**public** **void** updateSupplierName(Product product, String sname) {

product.setSupplierName(sname);

}

**public** **void** updateProductPrice(Product product, Integer p) {

product.setPrice(p);

}

}

## Question 4: Product 4

**import** java.util.ArrayList;

**import** java.util.List;

**import** java.util.Scanner;

**public** **class** Main {

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

Long id;

Integer price;

String pname, sname;

String choice;

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

List<Product> ob = **new** ArrayList<Product>();

Product objProduct;

ProductBO obj = **new** ProductBO();

**do** {

System.*out*.println("Enter the product id");

id = input.nextLong();

input.nextLine();

System.*out*.println("Enter the product name");

pname = input.nextLine();

System.*out*.println("Enter the supplier name");

sname = input.nextLine();

System.*out*.println("Enter the product price");

price = input.nextInt();

objProduct = **new** Product(id, pname, sname, price);

ob.add(objProduct);

System.*out*

.println("Do you want to enter the details of another product? Enter yes or no (not case sensitive)");

choice = input.next();

} **while** (choice.equalsIgnoreCase("YES"));

obj.displayAllProductDetails(ob);

}

}

**public** **class** Product {

**private** Long id;

**private** String productName;

**private** String supplierName;

**private** Integer price;

**public** Long getId() {

**return** id;

}

**public** **void** setId(Long id) {

**this**.id = id;

}

**public** String getProductName() {

**return** productName;

}

**public** **void** setProductName(String productName) {

**this**.productName = productName;

}

**public** String getSupplierName() {

**return** supplierName;

}

**public** **void** setSupplierName(String supplierName) {

**this**.supplierName = supplierName;

}

**public** Integer getPrice() {

**return** price;

}

**public** **void** setPrice(Integer price) {

**this**.price = price;

}

**public** Product() {

}

**public** Product(Long id, String productName, String supplierName,

Integer price) {

**this**.id = id;

**this**.productName = productName;

**this**.supplierName = supplierName;

**this**.price = price;

}

}

**import** java.util.Iterator;

**import** java.util.List;

**public** **class** ProductBO {

**public** **void** displayProductDetails(Product product) {

System.*out*.format("%-10s %-20s %-20s %-10s\n", product.getId(),

product.getProductName(), product.getSupplierName(),

product.getPrice());

}

**public** **void** displayAllProductDetails(Product productList[], **int** count) {

System.*out*.format("%-10s %-20s %-20s %-10s\n", "Id", "ProductName",

"SupplierName", "Price");

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

displayProductDetails(productList[i]);

}

}

**public** **void** displayAllProductDetails(List<Product> productList) {

System.*out*.format("%-10s %-20s %-20s %-10s\n", "Id", "ProductName",

"SupplierName", "Price");

**for** (Iterator<Product> it = productList.iterator(); it.hasNext();) {

displayProductDetails((it.next()));

}

}

**public** **void** updateProductName(Product product, String pname) {

product.setProductName(pname);

}

**public** **void** updateSupplierName(Product product, String sname) {

product.setSupplierName(sname);

}

**public** **void** updateProductPrice(Product product, Integer p) {

product.setPrice(p);

}

}

## Question 5: product 5

**import** java.util.ArrayList;

**import** java.util.List;

**import** java.util.Scanner;

**public** **class** Main {

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

Long id;

Integer price;

String pname, sname;

String choice;

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

List<Product> ob = **new** ArrayList<Product>();

Product objProduct;

ProductBO obj = **new** ProductBO();

**do** {

System.*out*.println("Enter the product id");

id = input.nextLong();

input.nextLine();

System.*out*.println("Enter the product name");

pname = input.nextLine();

System.*out*.println("Enter the supplier name");

sname = input.nextLine();

System.*out*.println("Enter the product price");

price = input.nextInt();

objProduct = **new** Product(id, pname, sname, price);

ob.add(objProduct);

System.*out*

.println("Do you want to enter the details of another product? Enter yes or no (not case sensitive)");

choice = input.next();

} **while** (choice.equalsIgnoreCase("YES"));

input.nextLine();

System.*out*.println("Enter the name of the product to be searched");

pname = input.nextLine();

obj.searchProductByName(ob, pname);

}

}

**public** **class** Product {

**private** Long id;

**private** String productName;

**private** String supplierName;

**private** Integer price;

**public** Long getId() {

**return** id;

}

**public** **void** setId(Long id) {

**this**.id = id;

}

**public** String getProductName() {

**return** productName;

}

**public** **void** setProductName(String productName) {

**this**.productName = productName;

}

**public** String getSupplierName() {

**return** supplierName;

}

**public** **void** setSupplierName(String supplierName) {

**this**.supplierName = supplierName;

}

**public** Integer getPrice() {

**return** price;

}

**public** **void** setPrice(Integer price) {

**this**.price = price;

}

**public** Product() {

}

**public** Product(Long id, String productName, String supplierName,

Integer price) {

**this**.id = id;

**this**.productName = productName;

**this**.supplierName = supplierName;

**this**.price = price;

}

}

**import** java.util.Iterator;

**import** java.util.List;

**public** **class** ProductBO {

**int** flag = 0;

**public** **void** displayProductDetails(Product product) {

System.*out*.format("%-10s %-20s %-20s %-10s\n", product.getId(),

product.getProductName(), product.getSupplierName(),

product.getPrice());

}

**public** **void** searchProductByName(List<Product> productList, String pname) {

Product object;

**for** (Iterator<Product> it = productList.iterator(); it.hasNext();) {

object = it.next();

**if** (object.getProductName().equalsIgnoreCase(pname)) {

System.*out*.format("%-10s %-20s %-20s %-10s\n", "Id",

"ProductName", "SupplierName", "Price");

displayProductDetails(object);

flag = 1;

}

}

**if** (flag == 0) {

System.*out*.println("No records found");

}

}

}

## Question 6: product 6

**import** java.util.ArrayList;

**import** java.util.List;

**import** java.util.Scanner;

**public** **class** Main {

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

Long id;

Integer price;

String pname, sname;

String choice;

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

List<Product> ob = **new** ArrayList<Product>();

Product objProduct;

ProductBO obj = **new** ProductBO();

**do** {

System.*out*.println("Enter the product id");

id = input.nextLong();

input.nextLine();

System.*out*.println("Enter the product name");

pname = input.nextLine();

System.*out*.println("Enter the supplier name");

sname = input.nextLine();

System.*out*.println("Enter the product price");

price = input.nextInt();

objProduct = **new** Product(id, pname, sname, price);

ob.add(objProduct);

System.*out*

.println("Do you want to enter the details of another product? Enter yes or no (not case sensitive)");

choice = input.next();

} **while** (choice.equalsIgnoreCase("YES"));

input.nextLine();

System.*out*.println("Enter the minimum price range");

**int** pl = input.nextInt();

System.*out*.println("Enter the maximum price range");

**int** ph = input.nextInt();

obj.searchProductByPriceRange(ob, pl, ph);

}

}

**public** **class** Product {

**private** Long id;

**private** String productName;

**private** String supplierName;

**private** Integer price;

**public** Long getId() {

**return** id;

}

**public** **void** setId(Long id) {

**this**.id = id;

}

**public** String getProductName() {

**return** productName;

}

**public** **void** setProductName(String productName) {

**this**.productName = productName;

}

**public** String getSupplierName() {

**return** supplierName;

}

**public** **void** setSupplierName(String supplierName) {

**this**.supplierName = supplierName;

}

**public** Integer getPrice() {

**return** price;

}

**public** **void** setPrice(Integer price) {

**this**.price = price;

}

**public** Product() {

}

**public** Product(Long id, String productName, String supplierName,

Integer price) {

**this**.id = id;

**this**.productName = productName;

**this**.supplierName = supplierName;

**this**.price = price;

}

}

**import** java.util.Iterator;

**import** java.util.List;

**public** **class** ProductBO {

**int** flag = 0;

**public** **void** displayProductDetails(Product product) {

System.*out*.format("%-10s %-20s %-20s %-10s\n", product.getId(),

product.getProductName(), product.getSupplierName(),

product.getPrice());

}

**public** **void** searchProductByPriceRange(List<Product> productList, **int** pl,

**int** ph) {

Product object;

**int** price;

**for** (Iterator<Product> it = productList.iterator(); it.hasNext();) {

object = it.next();

price = object.getPrice();

**if** (price >= pl && price <= ph) {

**if** (flag == 0) {

System.*out*.format("%-10s %-20s %-20s %-10s\n", "Id",

"ProductName", "SupplierName", "Price");

}

displayProductDetails(object);

flag = 1;

}

}

**if** (flag == 0) {

System.*out*.println("No records found");

}

}

**public** **void** searchProductByNameAndSupplier(List<Product> productList,

String pname, String sname) {

Product object;

**for** (Iterator<Product> it = productList.iterator(); it.hasNext();) {

object = it.next();

**if** (object.getProductName().equalsIgnoreCase(pname)

&& object.getSupplierName().equalsIgnoreCase(sname)) {

System.*out*.format("%-10s %-20s %-20s %-10s\n", "Id",

"ProductName", "SupplierName", "Price");

displayProductDetails(object);

flag = 1;

}

}

**if** (flag == 0) {

System.*out*.println("No records found");

}

}

}

## Question 7: product 7

**import** java.util.ArrayList;

**import** java.util.List;

**import** java.util.Scanner;

**public** **class** Main {

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

Long id;

Integer price;

String pname, sname;

String choice;

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

List<Product> ob = **new** ArrayList<Product>();

Product objProduct;

ProductBO obj = **new** ProductBO();

**do** {

System.*out*.println("Enter the product id");

id = input.nextLong();

input.nextLine();

System.*out*.println("Enter the product name");

pname = input.nextLine();

System.*out*.println("Enter the supplier name");

sname = input.nextLine();

System.*out*.println("Enter the product price");

price = input.nextInt();

objProduct = **new** Product(id, pname, sname, price);

ob.add(objProduct);

System.*out*

.println("Do you want to enter the details of another product? Enter yes or no (not case sensitive)");

choice = input.next();

} **while** (choice.equalsIgnoreCase("YES"));

input.nextLine();

System.*out*.println("Enter the minimum price range");

**int** pl = input.nextInt();

System.*out*.println("Enter the maximum price range");

**int** ph = input.nextInt();

obj.searchProductByPriceRange(ob, pl, ph);

}

}

**public** **class** Product {

**private** Long id;

**private** String productName;

**private** String supplierName;

**private** Integer price;

**public** Long getId() {

**return** id;

}

**public** **void** setId(Long id) {

**this**.id = id;

}

**public** String getProductName() {

**return** productName;

}

**public** **void** setProductName(String productName) {

**this**.productName = productName;

}

**public** String getSupplierName() {

**return** supplierName;

}

**public** **void** setSupplierName(String supplierName) {

**this**.supplierName = supplierName;

}

**public** Integer getPrice() {

**return** price;

}

**public** **void** setPrice(Integer price) {

**this**.price = price;

}

**public** Product() {

}

**public** Product(Long id, String productName, String supplierName,

Integer price) {

**this**.id = id;

**this**.productName = productName;

**this**.supplierName = supplierName;

**this**.price = price;

}

}

**import** java.util.Iterator;

**import** java.util.List;

**public** **class** ProductBO {

**int** flag = 0;

**public** **void** displayProductDetails(Product product) {

System.*out*.format("%-10s %-20s %-20s %-10s\n", product.getId(),

product.getProductName(), product.getSupplierName(),

product.getPrice());

}

**public** **void** searchProductByPriceRange(List<Product> productList, **int** pl,

**int** ph) {

Product object;

**int** price;

**for** (Iterator<Product> it = productList.iterator(); it.hasNext();) {

object = it.next();

price = object.getPrice();

**if** (price >= pl && price <= ph) {

**if** (flag == 0) {

System.*out*.format("%-10s %-20s %-20s %-10s\n", "Id",

"ProductName", "SupplierName", "Price");

}

displayProductDetails(object);

flag = 1;

}

}

**if** (flag == 0) {

System.*out*.println("No records found");

}

}

**public** **void** searchProductByNameAndSupplier(List<Product> productList,

String pname, String sname) {

Product object;

**for** (Iterator<Product> it = productList.iterator(); it.hasNext();) {

object = it.next();

**if** (object.getProductName().equalsIgnoreCase(pname)

&& object.getSupplierName().equalsIgnoreCase(sname)) {

System.*out*.format("%-10s %-20s %-20s %-10s\n", "Id",

"ProductName", "SupplierName", "Price");

displayProductDetails(object);

flag = 1;

}

}

**if** (flag == 0) {

System.*out*.println("No records found");

}

}

}

## Question 8: product 8

refer other class files in question 7

**import** java.util.ArrayList;

**import** java.util.Collection;

**import** java.util.Iterator;

**import** java.util.List;

**public** **class** ProductBO {

**int** flag = 0;

**public** **void** displayProductDetails(Product product) {

System.*out*.format("%-10s %-20s %-20s %-10s\n", product.getId(),

product.getProductName(), product.getSupplierName(),

product.getPrice());

}

**public** **void** displayAllProductDetails(List<Product> productList) {

System.*out*.format("%-10s %-20s %-20s %-10s\n", "Id", "ProductName",

"SupplierName", "Price");

**for** (Iterator<Product> it = productList.iterator(); it.hasNext();) {

displayProductDetails((it.next()));

}

}

**public** **void** deleteProduct(List<Product> productList, String pname) {

Product object;

Collection<Product> col = **new** ArrayList<Product>();

Iterator<Product> it = productList.iterator();

**while** (it.hasNext()) {

object = it.next();

**if** (object.getProductName().equals(pname)) {

col.add(object);

}

}

productList.removeAll(col);

}

}

# Object Oriented Programming III

## Question 1: Address Entity

**import** java.util.ArrayList;

**import** java.util.List;

**import** java.util.Scanner;

**public** **class** Main {

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

String street, city, district, state;

Integer doorNumber, pinCode;

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

Address obj = **new** Address();

List<Address> address = **new** ArrayList<Address>();

System.*out*.println("Enter the number of addresses to be inserted");

**int** num = in.nextInt();

**for** (**int** i = 1; i <= num; i++) {

System.*out*.println("Enter address " + i + " details");

System.*out*.println("Enter door number");

doorNumber = in.nextInt();

in.nextLine();

System.*out*.println("Enter street name");

street = in.nextLine();

System.*out*.println("Enter city name");

city = in.nextLine();

System.*out*.println("Enter district name");

district = in.nextLine();

System.*out*.println("Enter state name");

state = in.nextLine();

System.*out*.println("Enter pin code");

pinCode = in.nextInt();

in.nextLine();

address.add(**new** Address(doorNumber, street, city, district, state,

pinCode));

}

System.*out*.println("Enter a state name");

String sName = in.nextLine();

obj.listDistrict(address, sName);

System.*out*.println("Enter a district name");

String dName = in.nextLine();

obj.listCities(address, dName);

}

}

**import** java.util.Iterator;

**import** java.util.List;

**import** java.util.Set;

**import** java.util.TreeSet;

**public** **class** Address {

**private** Integer doorNumber;

**private** String street;

**private** String city;

**private** String district;

**private** String state;

**private** Integer pinCode;

**public** Integer getDoorNumber() {

**return** doorNumber;

}

**public** **void** setDoorNumber(Integer doorNumber) {

**this**.doorNumber = doorNumber;

}

**public** String getStreet() {

**return** street;

}

**public** **void** setStreet(String street) {

**this**.street = street;

}

**public** String getCity() {

**return** city;

}

**public** **void** setCity(String city) {

**this**.city = city;

}

**public** String getDistrict() {

**return** district;

}

**public** **void** setDistrict(String district) {

**this**.district = district;

}

**public** String getState() {

**return** state;

}

**public** **void** setState(String state) {

**this**.state = state;

}

**public** Integer getPinCode() {

**return** pinCode;

}

**public** **void** setPinCode(Integer pinCode) {

**this**.pinCode = pinCode;

}

**public** Address() {

}

**public** Address(Integer doorNumber, String street, String city,

String district, String state, Integer pinCode) {

**this**.doorNumber = doorNumber;

**this**.street = street;

**this**.city = city;

**this**.district = district;

**this**.state = state;

**this**.pinCode = pinCode;

}

**public** **void** listDistrict(List<Address> object, String stateName) {

**boolean** flag = **false**;

Address ob = **new** Address();

Set<String> sort = **new** TreeSet<String>();

**for** (Iterator<Address> it = object.iterator(); it.hasNext();) {

ob = it.next();

**if** (ob.state.equalsIgnoreCase(stateName)) {

**if** (flag == **false**) {

flag = **true**;

System.*out*.println("List of districts in " + ob.state

+ " are");

}

sort.add(ob.getDistrict());

}

}

**if** (flag == **false**) {

System.*out*.println("No records found");

} **else** {

**for** (Iterator<String> it = sort.iterator(); it.hasNext();) {

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

}

}

}

**public** **void** listCities(List<Address> object, String districtName) {

**boolean** flag = **false**;

Address ob = **new** Address();

Set<String> sort = **new** TreeSet<String>();

**for** (Iterator<Address> it = object.iterator(); it.hasNext();) {

ob = it.next();

**if** (ob.district.equalsIgnoreCase(districtName)) {

**if** (flag == **false**) {

flag = **true**;

System.*out*.println("List of cities in " + ob.district

+ " are");

}

sort.add(ob.getCity());

}

}

**if** (flag == **false**) {

System.*out*.println("No records found");

} **else** {

**for** (Iterator<String> it1 = sort.iterator(); it1.hasNext();) {

System.*out*.println(it1.next());

}

}

}

}

## Question 2: User Entity

**import** java.util.ArrayList;

**import** java.util.List;

**import** java.util.Scanner;

**public** **class** Main {

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

**int** count = 0;

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

Address obj = **new** Address();

User search = **new** User();

List<User> details = **new** ArrayList<User>();

System.*out*.println("Enter the number of user details to be inserted");

count = input.nextInt();

String uName;

String uNo, eMail, ccNO;

input.nextLine();

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

System.*out*.println("Enter details of user " + (i + 1));

System.*out*.println("Enter name");

uName = input.nextLine();

System.*out*.println("Enter mobile number");

uNo = input.nextLine();

System.*out*.println("Enter email address");

eMail = input.nextLine();

System.*out*.println("Enter address details");

System.*out*.println("Enter door number");

obj.setDoorNumber(input.nextInt());

input.nextLine();

System.*out*.println("Enter street name");

obj.setStreet(input.nextLine());

System.*out*.println("Enter city name");

obj.setCity(input.nextLine());

System.*out*.println("Enter district name");

obj.setDistrict(input.nextLine());

System.*out*.println("Enter state name");

obj.setState(input.nextLine());

System.*out*.println("Enter pin code");

obj.setPinCode(input.nextInt());

input.nextLine();

System.*out*.println("Enter credit card number");

ccNO = input.nextLine();

User data = **new** User(uName, uNo, eMail, obj, ccNO);

details.add(data);

}

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

System.*out*.println("1) Search for user details using mobile number.");

System.*out*

.println("2) Search for user details using mobile number and then change the email id of the above user to new id provided.");

System.*out*

.println("3) List the names of users in which the given string is a substring.");

System.*out*

.println("4) List the names of users from a particular district.");

System.*out*.println("Any other choice : Exit");

**boolean** flag = **false**;

**while** (flag == **false**) {

System.*out*.println("Enter your choice");

count = input.nextInt();

input.nextLine();

**switch** (count) {

**case** 1:

System.*out*.println("Enter the mobile number");

uNo = input.nextLine();

search.searchByMoblieNumber(details, uNo);

**break**;

**case** 2:

System.*out*.println("Enter the mobile number");

uNo = input.nextLine();

System.*out*.println("Enter the new email id");

eMail = input.nextLine();

search.searchAndChange(details, uNo, eMail);

**break**;

**case** 3:

System.*out*.println("Enter the substring");

uName = input.nextLine();

search.searchUserName(details, uName);

**break**;

**case** 4:

System.*out*.println("Enter the district name");

uName = input.nextLine();

search.searchNamesInDistrict(details, uName);

**break**;

**default**:

System.*out*.println("Application Terminated");

flag = **true**;

}

}

}

}

**import** java.util.Iterator;

**import** java.util.List;

**import** java.util.Set;

**import** java.util.TreeSet;

**public** **class** Address {

**private** Integer doorNumber;

**private** String street;

**private** String city;

**private** String district;

**private** String state;

**private** Integer pinCode;

**public** Integer getDoorNumber() {

**return** doorNumber;

}

**public** **void** setDoorNumber(Integer doorNumber) {

**this**.doorNumber = doorNumber;

}

**public** String getStreet() {

**return** street;

}

**public** **void** setStreet(String street) {

**this**.street = street;

}

**public** String getCity() {

**return** city;

}

**public** **void** setCity(String city) {

**this**.city = city;

}

**public** String getDistrict() {

**return** district;

}

**public** **void** setDistrict(String district) {

**this**.district = district;

}

**public** String getState() {

**return** state;

}

**public** **void** setState(String state) {

**this**.state = state;

}

**public** Integer getPinCode() {

**return** pinCode;

}

**public** **void** setPinCode(Integer pinCode) {

**this**.pinCode = pinCode;

}

**public** Address() {

}

**public** Address(Integer doorNumber, String street, String city,

String district, String state, Integer pinCode) {

**this**.doorNumber = doorNumber;

**this**.street = street;

**this**.city = city;

**this**.district = district;

**this**.state = state;

**this**.pinCode = pinCode;

}

**public** **void** listDistrict(List<Address> object, String stateName) {

**boolean** flag = **false**;

Address ob = **new** Address();

Set<String> sort = **new** TreeSet<String>();

**for** (Iterator<Address> it = object.iterator(); it.hasNext();) {

ob = it.next();

**if** (ob.state.equalsIgnoreCase(stateName)) {

**if** (flag == **false**) {

flag = **true**;

System.*out*.println("List of districts in " + ob.state

+ " are");

}

sort.add(ob.getDistrict());

}

}

**if** (flag == **false**) {

System.*out*.println("No records found");

} **else** {

**for** (Iterator<String> it = sort.iterator(); it.hasNext();) {

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

}

}

}

**public** **void** listCities(List<Address> object, String districtName) {

**boolean** flag = **false**;

Address ob = **new** Address();

Set<String> sort = **new** TreeSet<String>();

**for** (Iterator<Address> it = object.iterator(); it.hasNext();) {

ob = it.next();

**if** (ob.district.equalsIgnoreCase(districtName)) {

**if** (flag == **false**) {

flag = **true**;

System.*out*.println("List of cities in " + ob.district

+ " are");

}

sort.add(ob.getCity());

}

}

**if** (flag == **false**) {

System.*out*.println("No records found");

} **else** {

**for** (Iterator<String> it1 = sort.iterator(); it1.hasNext();) {

System.*out*.println(it1.next());

}

}

}

}

**import** java.util.Iterator;

**import** java.util.List;

**import** java.util.Set;

**import** java.util.TreeSet;

**public** **class** User {

**private** String name;

**private** String mobileNumber;

**private** String email;

**private** Address address;

**private** String creditCardNumber;

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** String getMobileNumber() {

**return** mobileNumber;

}

**public** **void** setMobileNumber(String mobileNumber) {

**this**.mobileNumber = mobileNumber;

}

**public** String getEmail() {

**return** email;

}

**public** **void** setEmail(String email) {

**this**.email = email;

}

**public** Address getAddress() {

**return** address;

}

**public** **void** setAddress(Address address) {

**this**.address = address;

}

**public** String getCreditCardNumber() {

**return** creditCardNumber;

}

**public** **void** setCreditCardNumber(String creditCardNumber) {

**this**.creditCardNumber = creditCardNumber;

}

**public** User(String name, String mobileNumber, String email,

Address address, String creditCardNumber) {

**this**.name = name;

**this**.mobileNumber = mobileNumber;

**this**.email = email;

**this**.address = address;

**this**.creditCardNumber = creditCardNumber;

}

**public** User() {

}

**public** **void** displayDetails(User object) {

System.*out*.println("User Details:");

System.*out*.println("Name : " + object.getName());

System.*out*.println("Email : " + object.getEmail());

System.*out*.println("Credit Card Number : "

+ object.getCreditCardNumber());

}

**public** **void** searchByMoblieNumber(List<User> details, String mobileNo) {

**boolean** flag = **false**;

User ob = **new** User();

**for** (Iterator<User> it = details.iterator(); it.hasNext();) {

ob = it.next();

**if** (ob.getMobileNumber().equalsIgnoreCase(mobileNo)) {

flag = **true**;

displayDetails(ob);

}

}

**if** (flag == **false**) {

System.*out*.println("No records found");

}

}

**public** **void** searchAndChange(List<User> details, String mobileNo,

String email) {

**boolean** flag = **false**;

User ob = **new** User();

//ArrayList<User> waste = new ArrayList<User>();

**for** (Iterator<User> it = details.iterator(); it.hasNext();) {

ob = it.next();

**if** (ob.getMobileNumber().equalsIgnoreCase(mobileNo)) {

flag = **true**;

ob.setEmail(email);

//details.add(ob);

}

}

**if** (flag == **false**) {

System.*out*.println("No records found");

} **else** {

// details.removeAll(waste);

}

}

**public** **void** searchUserName(List<User> details, String subStringName) {

**boolean** flag = **false**;

User ob = **new** User();

Set<String> names = **new** TreeSet<String>();

**for** (Iterator<User> it = details.iterator(); it.hasNext();) {

ob = it.next();

**if** (ob.getName().toLowerCase().contains(subStringName.toLowerCase())) {

flag = **true**;

names.add(ob.getName());

}

}

**if** (flag == **false**) {

System.*out*.println("No records found");

} **else** {

System.*out*.println("List of user names that contain the substring "

+ subStringName);

**for** (Iterator<String> it1 = names.iterator(); it1.hasNext();) {

System.*out*.println(it1.next());

}

}

}

**public** **void** searchNamesInDistrict(List<User> details, String dName) {

**boolean** flag = **false**;

User ob = **new** User();

Set<String> names1 = **new** TreeSet<String>();

**for** (Iterator<User> it = details.iterator(); it.hasNext();) {

ob = it.next();

**if** (ob.getAddress().getDistrict().equalsIgnoreCase(dName)) {

flag = **true**;

names1.add(ob.getName());

}

}

**if** (flag == **false**) {

System.*out*.println("No records found");

} **else** {

System.*out*.println("List of users from " + dName);

**for** (Iterator<String> it1 = names1.iterator(); it1.hasNext();) {

System.*out*.println(it1.next());

}

}

}

}

## Question 3: Author

**import** java.util.ArrayList;

**import** java.util.List;

**import** java.util.Scanner;

**public** **class** Main {

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

String name, email, gender;

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

Author data;

String ch;

List<Author> details = **new** ArrayList<Author>();

**do** {

System.*out*.println("Enter the author name");

name = input.nextLine();

System.*out*.println("Enter the author email id");

email = input.nextLine();

System.*out*.println("Enter the author's gender");

gender = input.nextLine();

System.*out*

.println("Do you want to add the details of another author? Type Yes/No (Not case sensitive)");

ch = input.next();

input.nextLine();

data = **new** Author(name, email, gender);

details.add(data);

} **while** (ch.equalsIgnoreCase("yes"));

AuthorBO obj = **new** AuthorBO();

obj.displayAllAuthorDetails(details);

System.*out*.println("Enter the name of the author to be searched");

name = input.nextLine();

data = obj.findAuthorByName(details, name);

**if** (data == **null**) {

System.*out*.println("Author not found");

} **else** {

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

}

obj.listAuthorNamesInSortedOrder(details);

}

}

**public** **class** Author **implements** Comparable<Author> {

**private** String name;

**private** String email;

**private** String gender;

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** String getEmail() {

**return** email;

}

**public** **void** setEmail(String email) {

**this**.email = email;

}

**public** String getGender() {

**return** gender;

}

**public** **void** setGender(String gender) {

**this**.gender = gender;

}

**public** Author(String name, String email, String gender) {

**this**.name = name;

**this**.email = email;

**this**.gender = gender;

}

**public** Author() {

}

@Override

**public** **int** compareTo(Author o) {

**return** **this**.getName().compareTo(o.getName());

}

**public** String toString() {

**return** **this**.getName() + " (" + **this**.getGender() + ") contact at "

+ **this**.getEmail();

}

}

**import** java.util.Iterator;

**import** java.util.List;

**import** java.util.Set;

**import** java.util.TreeSet;

**public** **class** AuthorBO {

**public** **void** displayAllAuthorDetails(List<Author> authorList) {

**boolean** flag = **false**;

Author object = **new** Author();

**for** (Iterator<Author> it = authorList.iterator(); it.hasNext();) {

object = it.next();

**if** (flag == **false**) {

System.*out*.format("%-20s %-20s %-20s\n", "Name", "Email",

"Gender");

flag = **true**;

}

System.*out*.format("%-20s %-20s %-20s\n", object.getName(),

object.getEmail(), object.getGender());

}

}

**public** **void** listAuthorNamesInSortedOrder(List<Author> authorList) {

Set<String> list = **new** TreeSet<String>();

**for** (Iterator<Author> it = authorList.iterator(); it.hasNext();) {

list.add(it.next().getName());

}

System.*out*.println("Alphabetically sorted author names");

**for** (Iterator<String> it = list.iterator(); it.hasNext();) {

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

}

}

**public** Author findAuthorByName(List<Author> authorList, String name) {

Author author = **new** Author();

**boolean** flag = **true**;

**for** (Iterator<Author> it = authorList.iterator(); it.hasNext() && flag;) {

author = it.next();

**if** (author.getName().equalsIgnoreCase(name)) {

flag = **false**;

}

}

**if**(flag==**true**){

author = **null**;

}

**return** author;

}

}

## Question 4: Java

**import** java.util.Iterator;

**import** java.util.List;

**public** **class** Book {

**private** String name;

**private** List<Author> authorList;

**private** **double** price;

**private** **int** qtyInStock = 0;

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** List<Author> getAuthorList() {

**return** authorList;

}

**public** **void** setAuthorList(List<Author> authorList) {

**this**.authorList = authorList;

}

**public** **double** getPrice() {

**return** price;

}

**public** **void** setPrice(**double** price) {

**this**.price = price;

}

**public** **int** getQtyInStock() {

**return** qtyInStock;

}

**public** **void** setQtyInStock(**int** qtyInStock) {

**this**.qtyInStock = qtyInStock;

}

**public** Book(String name, List<Author> authorList, **double** price,

**int** qtyInStock) {

**this**.name = name;

**this**.authorList = authorList;

**this**.price = price;

**this**.qtyInStock = qtyInStock;

}

**public** Book(String name, List<Author> authorList, **double** price) {

**this**.name = name;

**this**.authorList = authorList;

**this**.price = price;

**this**.qtyInStock = 0;

}

**public** Book() {

}

**public** String toString(){

String str1="",str2;

str2 = (**this**.qtyInStock>0)?"Available":"Not Available";

**for**(Iterator<Author> it = **this**.authorList.iterator();it.hasNext();){

str1+=it.next().getName()+" ";

}

**return** **this**.getName()+" authored by "+str1+"costs Rs."+**this**.getPrice()+" : "+str2;

}

}

**public** **class** Author **implements** Comparable<Author> {

**private** String name;

**private** String email;

**private** String gender;

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** String getEmail() {

**return** email;

}

**public** **void** setEmail(String email) {

**this**.email = email;

}

**public** String getGender() {

**return** gender;

}

**public** **void** setGender(String gender) {

**this**.gender = gender;

}

**public** Author(String name, String email, String gender) {

**this**.name = name;

**this**.email = email;

**this**.gender = gender;

}

**public** Author() {

}

@Override

**public** **int** compareTo(Author o) {

**return** **this**.getName().compareTo(o.getName());

}

**public** String toString() {

**return** **this**.getName() + " (" + **this**.getGender() + ") contact at "

+ **this**.getEmail();

}

}

**import** java.util.ArrayList;

**import** java.util.List;

**import** java.util.Scanner;

**public** **class** Main {

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

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

String name;

List<Author> authorList = **new** ArrayList<Author>();

**double** price;

**int** qtyInStock = 0;

**int** numAuthors;

String ans;

String authName;

String email;

String gender;

Book book;

System.*out*.println("Enter the book name");

name = in.nextLine();

System.*out*.println("Enter the number of authors");

numAuthors = in.nextInt();

in.nextLine();

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

System.*out*.println("Enter the author name");

authName = in.nextLine();

System.*out*.println("Enter the author email id");

email = in.nextLine();

System.*out*.println("Enter the author's gender");

gender = in.nextLine();

authorList.add(**new** Author(authName, email, gender));

}

System.*out*.println("Enter the book price");

price = in.nextDouble();

System.*out*

.println("Is the book currently available? Type Yes/No (Not case sensitive)");

ans = in.next();

**if** (ans.equalsIgnoreCase("yes")) {

System.*out*.println("Enter the number of books available");

qtyInStock = in.nextInt();

book = **new** Book(name, authorList, price, qtyInStock);

} **else** {

book = **new** Book(name, authorList, price);

}

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

}

}

# Exception Handling

## Question 1: Handle Exception

**import** java.util.Scanner;

**public** **class** Main {

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

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

**try** {

System.*out*.println("Enter an integer:");

String number = input.next();

**int** num = Integer.*parseInt*(number);

System.*out*.println("The square value is " + (num \* num));

System.*out*.println("The work has been done successfully");

} **catch** (NumberFormatException err) {

System.*out*

.println("Entered input is not a valid format for an integer.");

}

}

}

## Question 2: Exception handling 1

**import** java.util.Scanner;

**public** **class** Main {

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

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

**try** {

System.*out*.println("Enter the number of elements in the array");

**int** number = input.nextInt();

**int** array[] = **new** **int**[number];

System.*out*.println("Enter the elements in the array");

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

array[i] = input.nextInt();

}

System.*out*

.println("Enter the index of the array element you want to access");

number = input.nextInt();

System.*out*.println("The array element at index " + number + " = "

+ array[number]);

System.*out*.println("The array element successfully accessed");

} **catch** (ArrayIndexOutOfBoundsException err) {

System.*out*.println("java.lang.ArrayIndexOutOfBoundsException");

}

}

}

## Question 3: Exception handling 2

**import** java.io.BufferedReader;

**import** java.io.IOException;

**import** java.io.InputStreamReader;

**public** **class** Main {

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

BufferedReader in = **new** BufferedReader(**new** InputStreamReader(System.*in*));

**try** {

System.*out*.println("Enter the number of elements in the array");

String a = in.readLine();

**int** number = Integer.*parseInt*(a);

**int** array[] = **new** **int**[number];

System.*out*.println("Enter the elements in the array");

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

a = in.readLine();

array[i] = Integer.*parseInt*(a);

}

System.*out*

.println("Enter the index of the array element you want to access");

a = in.readLine();

number = Integer.*parseInt*(a);

System.*out*.println("The array element at index " + number + " = "

+ array[number]);

System.*out*.println("The array element successfully accessed");

} **catch** (ArrayIndexOutOfBoundsException err) {

System.*out*.println("java.lang.ArrayIndexOutOfBoundsException");

} **catch** (IOException err) {

System.*out*.println("java.lang.IOException");

} **catch** (NumberFormatException err) {

System.*out*.println("java.lang.NumberFormatException");

}

}

}

## Question 4: Exception Handling 3

**import** java.io.BufferedReader;

**import** java.io.IOException;

**import** java.io.InputStreamReader;

**public** **class** Main {

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

BufferedReader in = **new** BufferedReader(**new** InputStreamReader(System.*in*));

**try** {

System.*out*.println("Enter the number of elements in the array");

String a = in.readLine();

**int** number = Integer.*parseInt*(a);

**int** array[] = **new** **int**[number];

System.*out*.println("Enter the elements in the array");

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

a = in.readLine();

array[i] = Integer.*parseInt*(a);

}

System.*out*

.println("Enter the index of the array element you want to access");

a = in.readLine();

number = Integer.*parseInt*(a);

System.*out*.println("The array element at index " + number + " = "

+ array[number]);

System.*out*.println("The array element successfully accessed");

} **catch** (ArrayIndexOutOfBoundsException err) {

System.*out*.println("ArrayIndexOutOfBoundsException caught");

} **catch** (IOException err) {

System.*out*.println("IOException caught");

} **catch** (NumberFormatException err) {

System.*out*.println("NumberFormatException caught");

}

}

}

## Question 5: Exception Handling 4

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the 2 numbers");

**int** a = input.nextInt();

**int** b = input.nextInt();

**try** {

System.*out*

.println("The quotient of " + a + "/" + b + " = " + a / b);

} **catch** (ArithmeticException err) {

System.*out*.println("DivideByZeroException caught");

} **finally** {

System.*out*.println("Inside finally block");

}

}

}

## Question 6: Exception Handling 5

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter your age");

**int** number = input.nextInt();

**try** {

**if** (number <= 19) {

**throw** **new** InvalidAgeException(

"Exception occured: InvalidAgeException: not valid");

} **else** {

System.*out*.println("welcome to vote");

}

} **catch** (InvalidAgeException e) {

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

}

}

}

# FILE

## Question 1:

**import** java.io.File;

**import** java.io.FileInputStream;

**import** java.io.FileNotFoundException;

**import** java.io.FileOutputStream;

**import** java.io.FileReader;

**import** java.io.FileWriter;

**import** java.io.IOException;

**import** java.util.Scanner;

**public** **class** Main {

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

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

System.*out*.println("Enter the input file name");

String inputFile = in.nextLine();

System.*out*.println("Enter the output file name");

String outputFile = in.nextLine();

FileInputStream fis;

FileOutputStream fos;

**try** {

fis = **new** FileInputStream(**new** File(inputFile));

fos = **new** FileOutputStream(**new** File(outputFile));

**byte**[] buffer = **new** **byte**[1024];

**int** length;

**while** ((length = fis.read(buffer)) > 0) {

fos.write(buffer, 0, length);

}

} **catch** (FileNotFoundException e) {

e.printStackTrace();

} **catch** (IOException e) {

e.printStackTrace();

}

}

}

## Question 2: count

**import** java.io.File;

**import** java.io.FileNotFoundException;

**import** java.util.Scanner;

**public** **class** Main{

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

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

System.*out*.println("Enter the file name");

String inputFile = in.nextLine();

System.*out*.println("Enter the character to be counted");

String character = in.next();

String[] temp;

**int** count = 0;

**try**{

Scanner input = **new** Scanner(**new** File(inputFile));

**while**(input.hasNext()){

temp = input.next().split("");

**for**(String str : temp){

**if**(str.equalsIgnoreCase(character)){

count++;

}

}

}

System.*out*.println("File '"+inputFile+"' has "+count+" instances of letter '"+character+"'.");

}**catch**(FileNotFoundException e){

e.printStackTrace();

}

}

}

## Question 3:

import java.io.File;

import java.io.FileNotFoundException;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner in = new Scanner(System.in);

System.out.println("Enter the file name");

String inputFile = in.nextLine();

String temp;

try {

Scanner input = new Scanner(new File(inputFile));

while (input.hasNext()) {

temp = input.nextLine();

System.out.println(temp);

}

} catch (FileNotFoundException e) {

e.printStackTrace();

}

}

}

# Collections

## Question 1: ArrayList

**import** java.util.Iterator;

**import** java.util.Scanner;

**import** java.util.Set;

**import** java.util.TreeSet;

**public** **class** Main {

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

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

Set<String> details = **new** TreeSet<String>();

**int** count = 0;

String fName;

String lName;

String mobile;

String email;

String Address;

String data;

System.*out*.println("Enter The Number of Employees");

count = input.nextInt();

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

data = "";

System.*out*.println("Enter Employee " + (i + 1) + " Details:");

System.*out*.println("Enter the Firstname");

fName = input.next();

data += fName + " ";

System.*out*.println("Enter the Lastname");

lName = input.next();

data += lName + " ";

System.*out*.println("Enter the Mobile");

mobile = input.next();

data += mobile + " ";

System.*out*.println("Enter the Email");

email = input.next();

data += email + " ";

input.nextLine();

System.*out*.println("Enter the Address");

Address = input.nextLine();

data += Address;

details.add(data);

}

System.*out*.println("Employee List:");

System.*out*.format("%-15s %-15s %-15s %-30s %-15s\n", "Firstname",

"Lastname", "Mobile", "Email", "Address");

String array[];

**for** (Iterator<String> it = details.iterator(); it.hasNext();) {

data = it.next();

array = data.split(" ");

Address = "";

**for** (**int** i = 4; i < array.length; i++) {

Address += array[i] + " ";

}

Address = Address.substring(0, Address.length() - 1);

System.*out*.format("%-15s %-15s %-15s %-30s %-15s\n", array[0],

array[1], array[2], array[3], Address);

}

} }

## Question 2: Operatiions on string list

**import** java.util.ArrayList;

**import** java.util.Iterator;

**import** java.util.List;

**import** java.util.Scanner;

**public** **class** Main {

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

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

List<String> details = **new** ArrayList<String>();

**int** choice = 0;

String data;

**do** {

System.*out*.println("1. Insert");

System.*out*.println("2. Search");

System.*out*.println("3. Delete");

System.*out*.println("4. Display");

System.*out*.println("5. Exit");

System.*out*.println("Enter your choice :");

choice = Integer.*parseInt*(input.nextLine());

**switch** (choice) {

**case** 1:

System.*out*.println("Enter the item to be inserted:");

data = input.nextLine();

details.add(data);

System.*out*.println("Inserted successfully");

**break**;

**case** 2:

System.*out*.println("Enter the item to search :");

data = input.nextLine();

data = *listSearch*(details, data);

System.*out*.println(data);

**break**;

**case** 3:

System.*out*.println("Enter the item to delete :");

data = input.nextLine();

data = *listDelete*(details, data);

System.*out*.println(data);

**break**;

**case** 4:

*listDisplay*(details);

**break**;

**default**:

}

} **while** (choice > 0 && choice < 5);

}

**public** **static** String listSearch(List<String> list, String data) {

**boolean** flag = **false**;

**for** (Iterator<String> it = list.iterator(); it.hasNext();) {

**if** (data.equals(it.next())) {

flag = **true**;

}

}

data = (flag == **true**) ? "Item found in the list."

: "Item not found in the list.";

**return** data;

}

**public** **static** String listDelete(List<String> list, String data) {

List<String> del = **new** ArrayList<String>();

del.add(data);

**boolean** flag = **false**;

flag = list.removeAll(del);

data = (flag == **true**) ? "Deleted successfully" : "Item does not exist.";

**return** data;

}

**public** **static** **void** listDisplay(List<String> list) {

System.*out*.println("The Items in the list are :");

**for** (Iterator<String> it = list.iterator(); it.hasNext();) {

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

}

}

}

## Question 3: Collect Unique Symbols From Set of Card

**import** java.util.Iterator;

**import** java.util.Scanner;

**import** java.util.Set;

**import** java.util.TreeSet;

**class** Card **implements** Comparable<Card> {

String symbol;

String number;

**public** Card() {

}

**public** Card(String symbol, String number) {

**this**.symbol = symbol;

**this**.number = number;

}

@Override

**public** **int** compareTo(Card o) {

**return** **this**.symbol.compareTo(o.symbol);

}

}

**public** **class** Main {

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

**int** count = 0;

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

**boolean** flag = **false**;

Set<Card> list = **new** TreeSet<Card>();

String s, n;

**do** {

s = "";

n = "";

System.*out*.println("Enter a card :");

s += input.next().charAt(0);

n += input.next().charAt(0);

list.add(**new** Card(s, n));

count++;

flag = *check*(list);

} **while** (flag == **false**);

System.*out*.println("Four symbols gathered in " + count + " cards.");

System.*out*.println("Cards in Set are :");

Card c = **new** Card();

**for** (Iterator<Card> it = list.iterator(); it.hasNext();) {

c = it.next();

System.*out*.println(c.symbol + " " + c.number);

}

}

**public** **static** **boolean** check(Set<Card> list) {

**boolean** f = **false**;

**int** count = 0;

Set<String> data = **new** TreeSet<String>();

**for** (Iterator<Card> it = list.iterator(); it.hasNext();) {

count += (data.add(it.next().symbol)) ? 1 : 0;

}

f = (count == 4) ? **true** : **false**;

**return** f;

}

}

## Question 4: collect and group cards

**import** java.util.Iterator;

**import** java.util.ArrayList;

**import** java.util.Arrays;

**import** java.util.HashMap;

**import** java.util.List;

**import** java.util.Map;

**import** java.util.Scanner;

**import** java.util.Set;

**class** Card **implements** Comparable<Card> {

String symbol;

String number;

**public** Card() {

}

**public** Card(String symbol, String number) {

**this**.symbol = symbol;

**this**.number = number;

}

@Override

**public** **int** compareTo(Card o) {

**return** **this**.symbol.compareTo(o.symbol);

}

}

**public** **class** Main {

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

**int** count = 0;

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

Map<String, List<Card>> details = **new** HashMap<String, List<Card>>();

List<Card> l = **new** ArrayList<Card>();

System.*out*.println("Enter Number of Cards : ");

count = Integer.*parseInt*(input.nextLine());

String s, n;

Card c;

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

System.*out*.println("Enter card " + (i + 1) + ":");

s = "";

n = "";

s += input.next().charAt(0);

n += input.next();

c = **new** Card(s, n);

**if** (details.get(s) != **null**) {

// addMap(details, c, s);

l = **new** ArrayList<Card>();

l = details.get(s);

l.add(c);

} **else** {

l = **new** ArrayList<Card>();

l.add(c);

details.put(s, l);

}

}

Set<String> arr1 = details.keySet();

String[] arr = arr1.toArray(**new** String[0]);

Arrays.*sort*(arr);

System.*out*.println("Distinct Symbols are : ");

**for** (**int** i = 0; i < arr.length; i++) {

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

}

System.*out*.println();

**for** (**int** i = 0; i < arr.length; i++) {

**int** no = 0;

**int** sum = 0;

System.*out*.println("Cards in " + arr[i] + " Symbol");

List<Card> object = details.get(arr[i]);

**for** (Iterator<Card> it = object.iterator(); it.hasNext();) {

c = it.next();

System.*out*.println(c.symbol + " " + c.number);

no++;

sum += Integer.*parseInt*(c.number);

}

System.*out*.println("Number of cards : " + no);

System.*out*.println("Sum of Numbers : " + sum);

}

}

}

## Question 5: Set of Boxes

import java.text.DecimalFormat;

import java.util.Iterator;

import java.util.Scanner;

import java.util.Set;

import java.util.TreeSet;

public class Main {

public static void main(String[] args) {

int count = 0;

Scanner input = new Scanner(System.in);

Set<Box> details = new TreeSet<Box>();

Double length, width, height;

System.out.println("Enter the number of Box");

count = input.nextInt();

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

System.out.println("Enter the Box " + (i + 1) + " details");

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

length = input.nextDouble();

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

width = input.nextDouble();

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

height = input.nextDouble();

details.add(new Box(length, width, height));

}

System.out.print("Unique Boxes in the Set are");

Box obj;

for (Iterator<Box> it = details.iterator(); it.hasNext();) {

obj = it.next();

Double vol = obj.length \* obj.width \* obj.height;

DecimalFormat decimalFormat = new DecimalFormat("#.0#");

String result = decimalFormat.format(Double.valueOf(vol));

System.out.print("\nLength =" + (obj.length) + " Width ="

+ (obj.width) + " Height =" + (obj.height) + " Volume ="

+ result);

}

}

}

**public** **class** Box **implements** Comparable<Box> {

Double length;

Double width;

Double height;

**public** Box() {

}

**public** Box(Double len, Double wid, Double hei) {

**this**.length = len;

**this**.width = wid;

**this**.height = hei;

}

@Override

**public** **int** compareTo(Box o) {

**return** (**int**) ((**this**.length \* **this**.width \* **this**.height) - (o.length

\* o.width \* o.height));

}

**public** **boolean** equals(Box o){

**boolean** flag=**false**;

**if**(**this**.height\***this**.length\***this**.width == o.height\*o.length\*o.width){

flag = **true**;

}

**return** flag;

}

}

**import** java.text.DecimalFormat;

**public** **class** code {

/\*\*

\* **@param** args

\*/

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

// **TODO** Auto-generated method stub

Double num = 4.5000\*2.2\*23.1123;

DecimalFormat df = **new** DecimalFormat();

System.*out*.println("double : " + num);

System.*out*.println("double : " + df.format(num));

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

Double vol =num;

String str[] = vol.toString().split("");

str[1]= str[1].replaceAll("0", " ");

str[1]=str[1].trim();

str[1]=str[1].replaceAll(" ", "0");

System.*out*.println("double : " + str[0]);

System.*out*.println("double : " + num);

String s = "10.127777000";

DecimalFormat decimalFormat = **new** DecimalFormat("0.##");

String result = decimalFormat.format(Double.*valueOf*(s));

System.*out*.println(result);

}

}

# Multithreading

## Question 1: **Multi Threading**

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner in = new Scanner(System.in);

double x,y,z;

System.out.println("Enter the Degree for Sin : ");

x = in.nextDouble();

System.out.println("Enter the Degree for Cos : ");

y =in.nextDouble();

System.out.println("Enter the Degree for Tan : ");

z= in.nextDouble();

double p ;

SineClass a= new SineClass(x);

CosClass b = new CosClass(y);

TanClass c = new TanClass(z);

try{

a.t.join();

b.t.join();

c.t.join();

p = (a.s+b.s+c.s);

String format = String.format("%.2f", p);

System.out.print("Sum of sin, cos, tan = ");

System.out.println(format);

}catch(Exception e){

}

}

}

public class SineClass implements Runnable {

Double x;

Thread t;

Double s;

SineClass(Double x) {

this.x = x;

t = new Thread(this, "sinClass");

t.start();

}

public void run() {

try {

this.s = Math.sin(Math.toRadians(x));

Thread.sleep(1000);

} catch (InterruptedException e) {

}

}

}

public class CosClass implements Runnable {

Double x;

Thread t;

Double s;

CosClass(Double x) {

this.x = x;

t = new Thread(this, "CosClass");

t.start();

}

public void run() {

try {

this.s = Math.cos(Math.toRadians(x));

Thread.sleep(1000);

} catch (InterruptedException e) {

}

}

}

public class TanClass implements Runnable {

Double x;

Thread t;

Double s;

TanClass(Double x) {

this.x = x;

t = new Thread(this, "sinClass");

t.start();

}

// This is the entry point for thread.

public void run() {

try {

this.s = Math.tan(Math.toRadians(x));

Thread.sleep(1000);

} catch (InterruptedException e) {

}

}

}

## Question 2: Runnable Thread Vowel Counter and Shared Location

import java.util.Arrays;

import java.util.HashMap;

import java.util.Scanner;

import java.util.Set;

import java.util.logging.Level;

import java.util.logging.Logger;

import java.util.Iterator;

public class Main {

public static void main(String[] args) {

Scanner in = new Scanner(System.in);

String str = "";

int count = 0;

System.out.println("Enter Number of Counters :");

count = in.nextInt();

in.nextLine();

Counter a[] = new Counter[count] ;

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

System.out.println("Enter text for counter " + (i + 1) + " :");

str = in.nextLine();

a[i]= new Counter(str);

}

try {

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

a[i].t.join();

}

} catch (InterruptedException ex) {

Logger.getLogger(Main.class.getName()).log(Level.SEVERE, null, ex);

}

Set<Character> set= Counter.hs.keySet();

char arr[]= new char[set.size()];

int i=0;

char ch;

for(Iterator it = set.iterator();it.hasNext() ;i++){

ch = (Character) it.next();

arr[i]=ch;

}

Arrays.sort(arr);

System.out.println("Vowels count in given text are :");

for(i =0;i<arr.length;i++){

System.out.print(arr[i]+":"+Counter.hs.get(arr[i])+" ");

}

}

}

import java.util.Arrays;

import java.util.HashMap;

class Counter implements Runnable {

public static HashMap<Character,Integer> hs = new HashMap<Character, Integer>();

static{

Counter.hs.put('a', 0);

Counter.hs.put('e', 0);

Counter.hs.put('i', 0);

Counter.hs.put('o', 0);

Counter.hs.put('u', 0);

};

String data;

Thread t;

Double s;

Counter(String s) {

this.data = s.toLowerCase();

t = new Thread(this, "c");

t.start();

}

public static void addList(Character ch){

Integer count = hs.get(ch);

hs.put(ch, count+1);

}

public void run() {

try {

char ch;

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

ch = data.charAt(i);

switch(ch){

case 'a':

case 'e':

case 'i':

case 'o':

case 'u':

Counter.addList(ch);

break;

}

Thread.sleep(100);

}

} catch (InterruptedException e) {

}

}

}

## Question 3: **Character Frequency - Multiple Threads**

import java.util.Scanner;

import java.util.logging.Level;

import java.util.logging.Logger;

public class Main {

public static void main(String[] args) {

Scanner in = new Scanner(System.in);

String str = "";

int count = 0;

System.out.println("Enter Number of Counters :");

count = in.nextInt();

in.nextLine();

Counter a[] = new Counter[count] ;

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

System.out.println("Enter text for counter " + (i + 1) + " :");

str = in.next();

a[i]= new Counter(str);

}

try {

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

a[i].t.join();

}

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

System.out.println("Counter "+(i+1)+" Result :");

a[i].display();

}

} catch (InterruptedException ex) {

Logger.getLogger(Main.class.getName()).log(Level.SEVERE, null, ex);

}

}

}

import java.util.Arrays;

import java.util.HashMap;

import java.util.Iterator;

import java.util.Set;

public class Counter extends Thread {

String data;

public HashMap<Character,Integer> hs = new HashMap<Character, Integer>();

Thread t;

Double s;

Counter(String s) {

this.data = s;

t = new Thread(this, "c");

t.start();

}

public void addList(Character ch){

Integer count = this.hs.get(ch);

count=(count==null)?1:count+1;

this.hs.put(ch, count);

}

@Override

public void run() {

try {

char ch;

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

ch = data.charAt(i);

if(Character.isAlphabetic(ch)){

this.addList(ch);

}

}

Thread.sleep(100);

} catch (InterruptedException e) {

}

}

public synchronized void display(){

Set<Character> set= this.hs.keySet();

char arr[]= new char[set.size()];

int i=0;

char ch;

for(Iterator it = set.iterator();it.hasNext() ;i++){

ch = (Character) it.next();

arr[i]=ch;

}

Arrays.sort(arr);

for(i =0;i<arr.length;i++){

System.out.print(arr[i]+":"+this.hs.get(arr[i])+" ");

}

System.out.println();

}

}

# CRUD

## Question 1: JDBC Contact Insert

**import** java.sql.\*;

**import** java.util.Scanner;

**public** **class** Main {

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

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

Connection con = ContactDAO.*connect*();

Contact O = **new** Contact();

ContactBO sql = **new** ContactBO();

sql.getConnection(con);

System.*out*.println("Before the insert");

sql.display(con);

System.*out*.println("Enter Contact Table Inputs");

System.*out*.println("Enter The FirstName :");

O.setFirstName(input.next());

System.*out*.println("Enter The LastName :");

O.setLastName(input.next());

System.*out*.println("Enter The Email :");

O.setEmail(input.next());

System.*out*.println("Enter The PhoneNumber :");

O.setPhoneNo(input.next());

System.*out*.println("After the insert");

sql.insert(O, con);

sql.display(con);

ContactDAO.*closeConnection*();

}

}

**import** java.sql.Connection;

**import** java.sql.DriverManager;

**import** java.sql.SQLException;

**import** java.util.ResourceBundle;

**public** **class** ContactDAO {

**static** Connection *connection*;

**public** **static** Connection connect() {

ResourceBundle rb = ResourceBundle.*getBundle*("mysql");

String url = rb.getString("db.url");

String username = rb.getString("db.username");

String password = rb.getString("db.password");

**try** {

Class.*forName*("com.mysql.jdbc.Driver");

*connection* = DriverManager.*getConnection*(url, username, password);

} **catch** (SQLException e) {

e.printStackTrace();

} **catch** (ClassNotFoundException e) {

e.printStackTrace();

}

**return** *connection*;

}

**public** **static** **void** closeConnection() {

**try** {

*connection*.close();

} **catch** (SQLException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

}

**import** java.sql.Connection;

**import** java.sql.PreparedStatement;

**import** java.sql.ResultSet;

**import** java.sql.SQLException;

**import** java.sql.Statement;

**public** **class** ContactBO {

Statement stmt = **null**;

**static** Connection *connect*;

**static** String *insertQuery* = "insert into contact(firstName,lastName,email,phoneNo) values(?,?,?,?)";

**static** String *displayQuery* = "select \* from contact";

Statement statement = **null**;

ResultSet resultset = **null**;

**static** PreparedStatement *pStatement*;

**public** **void** insert(Contact O, Connection con) {

**try** {

*pStatement* = con.prepareStatement(*insertQuery*);

*pStatement*.setString(1, O.getFirstName());

*pStatement*.setString(2, O.getLastName());

*pStatement*.setString(3, O.getEmail());

*pStatement*.setString(4, O.getPhoneNo());

*pStatement*.executeUpdate(); // inserting of record

statement.executeQuery("commit");

} **catch** (SQLException e) {

e.printStackTrace();

}

}

**public** **void** getConnection(Connection connect) {

**try** {

*pStatement* = connect.prepareStatement(*insertQuery*);

statement = connect.createStatement();

} **catch** (SQLException e) {

e.printStackTrace();

}

}

**public** **void** display(Connection connect) {

String fName, lName, email, phoneNo;

**try** {

Statement statement1 = connect.createStatement();

resultset = statement1.executeQuery(*displayQuery*);

System.*out*.format("%-15s %-15s %-30s %-15s\n", "FirstName",

"LastName", "Email", "PhoneNumber");

// resultset.next();

// System.out.println(resultset.getString(1));

**while** (resultset.next()) {

fName = resultset.getString(2);

lName = resultset.getString(3);

email = resultset.getString(4);

phoneNo = resultset.getString(5);

System.*out*.format("%-15s %-15s %-30s %-15s\n", fName, lName,

email, phoneNo);

}

} **catch** (SQLException e) {

e.printStackTrace();

}

}

}

**public** **class** Contact {

**private** String firstName;

**private** String lastName;

**private** String email;

**private** String phoneNo;

**public** String getFirstName() {

**return** firstName;

}

**public** **void** setFirstName(String firstName) {

**this**.firstName = firstName;

}

**public** String getLastName() {

**return** lastName;

}

**public** **void** setLastName(String lastName) {

**this**.lastName = lastName;

}

**public** String getEmail() {

**return** email;

}

**public** **void** setEmail(String email) {

**this**.email = email;

}

**public** String getPhoneNo() {

**return** phoneNo;

}

**public** **void** setPhoneNo(String phoneNo) {

**this**.phoneNo = phoneNo;

}

**public** Contact(String firstName, String lastName, String email,

String phoneNo) {

**this**.firstName = firstName;

**this**.lastName = lastName;

**this**.email = email;

**this**.phoneNo = phoneNo;

}

**public** Contact() {

}

}

## Question 2: JDBC –list

**import** java.sql.\*;

**public** **class** Main {

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

Connection con = DbConnector.*connect*();

ContactBO sql = **new** ContactBO();

sql.getConnection(con);

sql.display(con);

DbConnector.*closeConnection*();

}

}

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

import java.util.ResourceBundle;

public class DbConnector {

static Connection connection;

public static Connection connect() {

ResourceBundle rb = ResourceBundle.getBundle("mysql");

String url = rb.getString("db.url");

String username = rb.getString("db.username");

String password = rb.getString("db.password");

try {

Class.forName("com.mysql.jdbc.Driver");

connection = DriverManager.getConnection(url, username, password);

} catch (SQLException e) {

e.printStackTrace();

} catch (ClassNotFoundException e) {

e.printStackTrace();

}

return connection;

}

public static void closeConnection() {

try {

connection.close();

} catch (SQLException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

}

**import** java.sql.Connection;

**import** java.sql.PreparedStatement;

**import** java.sql.ResultSet;

**import** java.sql.SQLException;

**import** java.sql.Statement;

**public** **class** ContactBO {

Statement stmt = **null**;

**static** Connection *connect*;

**static** String *insertQuery* = "insert into contact(firstName,lastName,email,phoneNo) values(?,?,?,?)";

**static** String *displayQuery* = "select \* from contact";

Statement statement = **null**;

ResultSet resultset = **null**;

**static** PreparedStatement *pStatement*;

**public** **void** insert(Contact O, Connection con) {

**try** {

*pStatement* = con.prepareStatement(*insertQuery*);

*pStatement*.setString(1, O.getFirstName());

*pStatement*.setString(2, O.getLastName());

*pStatement*.setString(3, O.getEmail());

*pStatement*.setString(4, O.getPhoneNo());

*pStatement*.executeUpdate(); // inserting of record

statement.executeQuery("commit");

} **catch** (SQLException e) {

e.printStackTrace();

}

}

**public** **void** getConnection(Connection connect) {

**try** {

*pStatement* = connect.prepareStatement(*insertQuery*);

statement = connect.createStatement();

} **catch** (SQLException e) {

e.printStackTrace();

}

}

**public** **void** display(Connection connect) {

String fName, lName, email, phoneNo;

**try** {

Statement statement1 = connect.createStatement();

resultset = statement1.executeQuery(*displayQuery*);

System.*out*.format("%-15s %-15s %-30s %-15s\n", "Firstname",

"Lastname", "E-Mail", "Phone Number");

// resultset.next();

// System.out.println(resultset.getString(1));

**while** (resultset.next()) {

fName = resultset.getString(2);

lName = resultset.getString(3);

email = resultset.getString(4);

phoneNo = resultset.getString(5);

System.*out*.format("%-15s %-15s %-30s %-15s\n", fName, lName,

email, phoneNo);

}

} **catch** (SQLException e) {

e.printStackTrace();

}

}

}

**public** **class** Contact {

**private** String firstName;

**private** String lastName;

**private** String email;

**private** String phoneNo;

**public** String getFirstName() {

**return** firstName;

}

**public** **void** setFirstName(String firstName) {

**this**.firstName = firstName;

}

**public** String getLastName() {

**return** lastName;

}

**public** **void** setLastName(String lastName) {

**this**.lastName = lastName;

}

**public** String getEmail() {

**return** email;

}

**public** **void** setEmail(String email) {

**this**.email = email;

}

**public** String getPhoneNo() {

**return** phoneNo;

}

**public** **void** setPhoneNo(String phoneNo) {

**this**.phoneNo = phoneNo;

}

**public** Contact(String firstName, String lastName, String email,

String phoneNo) {

**this**.firstName = firstName;

**this**.lastName = lastName;

**this**.email = email;

**this**.phoneNo = phoneNo;

}

**public** Contact() {

}

}

## Question 3: JDBC - Update phoneNo

import java.sql.\*;

import java.util.Scanner;

public class Main {

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

Scanner input = new Scanner(System.in);

Connection con = ContactDAO.connect();

ContactBO sql = new ContactBO();

sql.getConnection(con);

System.out.println("Updating The PhoneNumber of the Contact");

System.out

.println("Enter The FirstName of the Contact to update his PhoneNumber");

String name = input.next();

System.out.println("Before the update");

sql.display(con);

System.out.println("Enter The PhoneNumber of the Contact");

String phone = input.next();

sql.updateRecord(con, name, phone);

System.out.println("After the update");

sql.display(con);

ContactDAO.closeConnection();

}

}

**import** java.sql.Connection;

**import** java.sql.DriverManager;

**import** java.sql.SQLException;

**import** java.util.ResourceBundle;

**public** **class** ContactDAO {

**static** Connection *connection*;

**public** **static** Connection connect() {

ResourceBundle rb = ResourceBundle.*getBundle*("mysql");

String url = rb.getString("db.url");

String username = rb.getString("db.username");

String password = rb.getString("db.password");

**try** {

Class.*forName*("com.mysql.jdbc.Driver");

*connection* = DriverManager.*getConnection*(url, username, password);

} **catch** (SQLException e) {

e.printStackTrace();

} **catch** (ClassNotFoundException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

**return** *connection*;

}

**public** **static** **void** closeConnection() {

**try** {

*connection*.close();

} **catch** (SQLException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

}

**public** **class** Contact {

**private** String firstName;

**private** String lastName;

**private** String email;

**private** String phoneNo;

**public** String getFirstName() {

**return** firstName;

}

**public** **void** setFirstName(String firstName) {

**this**.firstName = firstName;

}

**public** String getLastName() {

**return** lastName;

}

**public** **void** setLastName(String lastName) {

**this**.lastName = lastName;

}

**public** String getEmail() {

**return** email;

}

**public** **void** setEmail(String email) {

**this**.email = email;

}

**public** String getPhoneNo() {

**return** phoneNo;

}

**public** **void** setPhoneNo(String phoneNo) {

**this**.phoneNo = phoneNo;

}

**public** Contact(String firstName, String lastName, String email,

String phoneNo) {

**this**.firstName = firstName;

**this**.lastName = lastName;

**this**.email = email;

**this**.phoneNo = phoneNo;

}

**public** Contact() {

}

}

**import** java.sql.Connection;

**import** java.sql.PreparedStatement;

**import** java.sql.ResultSet;

**import** java.sql.SQLException;

**import** java.sql.Statement;

**public** **class** ContactBO {

Statement stmt = **null**;

**static** Connection *connect*;

**static** String *insertQuery* = "insert into contact(firstName,lastName,email,phoneNo) values(?,?,?,?)";

**static** String *displayQuery* = "select \* from contact";

**static** String *searchQuery* = "select \* from contact where fName=?";

**static** String *updateQuery* = "update contact set phoneNo=? where firstName=?";

Statement statement = **null**;

ResultSet resultset = **null**;

**static** PreparedStatement *pStatement*;

**public** **void** updateRecord(Connection connect, String fname, String Phone) {

**try** {

*pStatement* = connect.prepareStatement(*updateQuery*);

*pStatement*.setString(2, fname);

*pStatement*.setString(1, Phone);

*pStatement*.executeUpdate();

} **catch** (SQLException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

**public** **void** getConnection(Connection connect) {

**try** {

*pStatement* = connect.prepareStatement(*insertQuery*);

statement = connect.createStatement();

} **catch** (SQLException e) {

e.printStackTrace();

}

}

**public** **void** display(Connection connect) {

String fName, lName, email, phoneNo;

**try** {

Statement statement1 = connect.createStatement();

resultset = statement1.executeQuery(*displayQuery*);

System.*out*.format("%-15s %-15s %-30s %-15s\n", "FirstName",

"LastName", "Email", "PhoneNumber");

// resultset.next();

// System.out.println(resultset.getString(1));

**while** (resultset.next()) {

fName = resultset.getString(2);

lName = resultset.getString(3);

email = resultset.getString(4);

phoneNo = resultset.getString(5);

System.*out*.format("%-15s %-15s %-30s %-15s\n", fName, lName,

email, phoneNo);

}

} **catch** (SQLException e) {

e.printStackTrace();

}

}

}

## Question 4: JDBC – Menu Driven

**import** java.util.Scanner;

**public** **class** Student {

**int** rollNo;

String name;

**int** age;

**int** mark;

String department;

**public** **int** getRollNo() {

**return** rollNo;

}

**public** **void** setRollNo(**int** rollNo) {

**this**.rollNo = rollNo;

}

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** **int** getAge() {

**return** age;

}

**public** **void** setAge(**int** age) {

**this**.age = age;

}

**public** **int** getMark() {

**return** mark;

}

**public** **void** setMark(**int** mark) {

**this**.mark = mark;

}

**public** String getDepartment() {

**return** department;

}

**public** **void** setDepartment(String department) {

**this**.department = department;

}

**public** Student(**int** rollNo, String name, **int** age, **int** mark, String department) {

**this**.rollNo = rollNo;

**this**.name = name;

**this**.age = age;

**this**.mark = mark;

**this**.department = department;

}

**public** Student() {

}

**public** **void** getData( Scanner input) {

//Scanner input = new Scanner(System.in);

System.*out*.println("Enter the Roll Number :");

**this**.rollNo = input.nextInt();

System.*out*.println("Enter the Name :");

**this**.name = input.next();

System.*out*.println("Enter the Age :");

**this**.age = input.nextInt();

System.*out*.println("Enter the Total marks :");

**this**.mark = input.nextInt();

System.*out*.println("Enter the Department :");

**this**.department = input.next();

}

}

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

import java.util.ResourceBundle;

public class ContactDAO {

static Connection connection;

public static Connection connect() {

ResourceBundle rb = ResourceBundle.getBundle("mysql");

String url = rb.getString("db.url");

String username = rb.getString("db.username");

String password = rb.getString("db.password");

try {

Class.forName("com.mysql.jdbc.Driver");

connection = DriverManager.getConnection(url, username, password);

} catch (SQLException e) {

e.printStackTrace();

} catch (ClassNotFoundException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

return connection;

}

public static void closeConnection() {

try {

connection.close();

} catch (SQLException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

}

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

public class ContactBO {

Statement stmt = null;

static Connection connection;

static String insertQuery = "insert into student values(?,?,?,?,?)";

static String displayQuery = "select \* from student";

static String deleteQuery = "delete from student where RollNo=?";

static String updateQuery = "update student set xxx =? where RollNo=?";

Statement statement = null;

ResultSet resultset = null;

static PreparedStatement pStatement;

public void getConnection(Connection connect) {

connection = connect;

}

public boolean updateRecord(String rollNo, String field, String data) {

int flag = 0;

try {

updateQuery=updateQuery.replaceFirst("xxx", field.toString());

//System.out.println(updateQuery);

pStatement = connection.prepareStatement(updateQuery);

// pStatement.setString(1, field.toString());

if (field.equalsIgnoreCase("Age") || field.equalsIgnoreCase("Mark")) {

pStatement.setInt(1, Integer.parseInt(data));

} else {

pStatement.setString(1, data);

}

pStatement.setLong(2, Integer.parseInt(rollNo));

flag = pStatement.executeUpdate();

} catch (SQLException e) {

e.printStackTrace();

}

return (flag == 0) ? false : true;

}

public boolean insert(Student O) {

int flag = 0;

try {

pStatement = connection.prepareStatement(insertQuery);

pStatement.setString(1, Integer.toString(O.getRollNo()));

pStatement.setString(2, O.getName());

pStatement.setString(3, Integer.toString(O.getAge()));

pStatement.setString(4, Integer.toString(O.getMark()));

pStatement.setString(5, O.getDepartment());

flag = pStatement.executeUpdate();

} catch (SQLException e) {

e.printStackTrace();

}

return (flag == 0) ? false : true;

}

public boolean delete(String roll) {

int flag = 0;

try {

pStatement = connection.prepareStatement(deleteQuery);

pStatement.setInt(1,Integer.parseInt(roll));

flag = pStatement.executeUpdate();

} catch (SQLException e) {

e.printStackTrace();

}

return (flag == 0) ? false : true;

}

public void display() {

try {

pStatement = connection.prepareStatement(displayQuery);

resultset = pStatement.executeQuery();

while (resultset.next()) {

System.out.println("Roll No :" + resultset.getInt("RollNo"));

System.out.println("Name :" + resultset.getString("name"));

System.out.println("Age :" + resultset.getString("age"));

System.out.println("Total marks :"

+ resultset.getString("mark"));

System.out.println("Department :"

+ resultset.getString("department"));

System.out.println();

}

} catch (SQLException e) {

e.printStackTrace();

}

}

}

import java.sql.\*;

import java.util.Scanner;

public class Main {

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

Scanner input = new Scanner(System.in);

Connection con = ContactDAO.connect();

ContactBO sql = new ContactBO();

boolean f = true;

sql.getConnection(con);

do {

System.out.println("MENU");

System.out.println("1.Insert");

System.out.println("2.Update");

System.out.println("3.Delete");

System.out.println("4.Display");

System.out.println("5.Exit");

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

int ch = input.nextInt();

switch (ch) {

case 1:

Student obj = new Student();

obj.getData(input);

sql.insert(obj);

System.out.println("Record Inserted");

break;

case 2:

update(sql,input);

break;

case 3:

System.out

.println("Enter the Roll Number of the student to be deleted :");

String roll = input.next();

boolean flag = sql.delete(roll);

if (flag == false) {

System.out.println("No Record Found");

} else {

System.out.println("Record Deleted Successfully");

}

case 4:

sql.display();

break;

default:

f = false;

break;

}

} while (f);

ContactDAO.closeConnection();

}

public static void update(ContactBO obj,Scanner in) {

System.out

.println("Enter the Roll Number of the student to be updated :");

String roll = in.next();

boolean flag=false;

if(roll.length()>2){

System.out.println("Do you want to update :");

System.out.println("1.Name");

System.out.println("2.Age");

System.out.println("3.Total Marks");

System.out.println("4.Department");

System.out.println("5.Exit");

System.out.println("Enter Your choice :");

int ch = in.nextInt();

String name, dept;

int age, total;

String temp="a",field="b";

switch (ch) {

case 1:

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

name = in.next();

temp = name;

field = "Name";

break;

case 2:

System.out.println("Enter the Age :");

temp = Integer.toString(in.nextInt());

field = "Age";

break;

case 3:

System.out.println("Enter the Total Marks :");

dept = in.next();

temp = Integer.toString(in.nextInt());

field = "Mark";

break;

case 4:

System.out.println("Enter the Department :");

dept = in.next();

temp = dept;

field = "Department";

break;

default:

}

flag = obj.updateRecord(roll, field, temp);

}

if (flag) {

System.out.println("Record Updated Successfully");

} else {

System.out.println("No Record Found");

}

}

}