**package** august;

**import** java.util.\*;

//--------------------------------------------------------------------------------------------------

/\*class ZeroException extends Exception

{ public String toString()

{

return "no is less than or equal to 0";

}

}\*/

//--------------------------------------------------------------------------------------------------

/\*class NameException extends Exception{

public String toString(){

return "name is invalid";

}

}\*/

//--------------------------------------------------------------------------------------------------

/\*class AgeNotWithinRangeException extends Exception{

public String toString(){

return "Age is not in range";

}

}

class NameNotValidException extends Exception{

public String toString(){

return "name is invalid";

}

}\*/

//--------------------------------------------------------------------------------------------------

/\*class student{

int rollno,age; String name,course;

student(int rollno,int age, String name,String course){

this.rollno=rollno;

this.age=age;

this.name=name;

this.course=course;

}

void display(){

System.out.println("Roll No "+rollno+"\tName "+name+"\tAge "+age+"\tCourse "+course);

}

}\*/

//--------------------------------------------------------------------------------------------------

/\*class VowelException extends Exception{

public String toString() {

return "Character is Vowel";

}

}

class BlankException extends Exception{

public String toString() {

return "Character is Blank";

}

}

class ExitException extends Exception{

public String toString() {

return "Character is an Exit Character";

}

}

class Test{

Scanner sc=new Scanner(System.in);

char ch;

Test(){

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

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

}

void display() {

try {

if(ch=='a' || ch=='e' || ch=='i' || ch=='o' || ch=='u' || ch=='A' || ch=='E' || ch=='I' || ch=='O' || ch=='U') {

throw new VowelException();

}

else if(ch==' ')

throw new BlankException();

else if(ch=='X')

throw new ExitException();

else

System.out.println("Valid Character");

}

catch(VowelException e) {

System.out.println(e);

}

catch(BlankException e) {

System.out.println(e);

}

catch(ExitException e) {

System.out.println(e);

}

}

}\*/

//--------------------------------------------------------------------------------------------------

/\*class DayException extends Exception

{

public String toString()

{

return "Day is not valid";

}

}

class MonthException extends Exception

{

public String toString()

{

return "Month is invalid";

}

}

class YearException extends Exception

{

public String toString()

{

return "Year is invalid";

}

}\*/

//-------------------------------------------------------------------------------------------------------

/\*class InvalidOperatorException extends Exception{

public String toString() {

return "Invalid Operator";

}

}

class NegativeResultException extends Exception{

public String toString() {

return "Result is Negative";

}

}

class Calculator{

double n1,n2;

char operator;

Calculator(double n1,double n2,char ch){

this.n1=n1;

this.n2=n2;

operator=ch;

}

void add(){

try {

if(n1<0 || n2<0)

throw new NegativeResultException();

else

System.out.println("Addition "+(n1+n2));

}

catch(NegativeResultException e) {

System.out.println(e);

}

}

void subtract() {

try {

double r=n1-n2;

if(r<0)

throw new NegativeResultException();

else

System.out.println("Subtraction "+r);

}

catch(NegativeResultException e) {

System.out.println(e);

}

}

void mul() {

try {

if((n1<0 && n2>0) || (n1>0 && n2<0))

throw new NegativeResultException();

else

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

}

catch(NegativeResultException e) {

System.out.println(e);

}

}

void div() {

try {

if(n1<0 || n2<0)

throw new NegativeResultException();

else

System.out.println("Division "+(n1/n2));

}

catch(NegativeResultException e) {

System.out.println(e);

}

}

}\*/

//------------------------------------------------------------------------------------------------------------------

/\*class NotAnAlphabet extends Exception{

public String toString(){

return "Character is not an Alphabet";

}

}

class Error extends Exception{

public String toString(){

return "Error: Not an rainbow colour";

}

}

class DisplayColor{

Scanner sc=new Scanner(System.in);

char ch;

DisplayColor(){

System.out.println("Enter an alphabet");

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

}

void display() {

try {

if(Character.isAlphabetic(ch)) {

if(ch=='v' || ch=='V')

System.out.println("Violet");

else if(ch=='i' || ch=='I')

System.out.println("Indigo");

else if(ch=='b' || ch=='B')

System.out.println("Blue");

else if(ch=='g' || ch=='G')

System.out.println("Green");

else if(ch=='o' || ch=='O')

System.out.println("Orange");

else if(ch=='y' || ch=='Y')

System.out.println("Yellow");

else if(ch=='r' || ch=='R')

System.out.println("Red");

else

throw new Error();

}

else

throw new NotAnAlphabet();

}

catch(NotAnAlphabet e) {

System.out.println(e);

}

catch(Error e) {

System.out.println(e);

}

}

}\*/

//----------------------------------------------------------------------------------------------------------------

**class** InvalidUsernameException **extends** Exception{

**public** String toString() {

**return** "Invalid Username Exception";

}

}

**class** InvalidPasswordException **extends** Exception{

**public** String toString() {

**return** "Invalid Password Exception";

}

}

**class** Email{

String username,password;

Email(){

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

System.***out***.println("Enter username, password");

username=sc.next();

password=sc.next();

}

Email(String u,String p){

username=u;

password=p;

}

**void** display() {

**try** {

**if**(!username.equals("admin"))

**throw** **new** InvalidUsernameException();

**if**(!password.equals("abc@123"))

**throw** **new** InvalidPasswordException();

System.***out***.println("Logged In Successfully");

}

**catch**(InvalidUsernameException e) {

System.***out***.println(e);

}

**catch**(InvalidPasswordException e) {

System.***out***.println(e);

}

}

}

//----------------------------------------------------------------------------------------------------------------

/\*class Positive extends Exception{

public String toString(){

return "Patient is Covid Positive(+) and Need to Hospitalized";

}

}

class Patient{

Scanner sc=new Scanner(System.in);

int age,oxyLevel,hrtcReport;

String name;

Patient(){

System.out.println("Enter name, age, oxygen level, hrtc report");

name=sc.next();

age=sc.nextInt();

oxyLevel=sc.nextInt();

hrtcReport=sc.nextInt();

}

void display() {

System.out.println("Name "+name+"\tAge "+age+"\tOxygen level"+oxyLevel+"\tHRTC Report "+hrtcReport);

}

}\*/

**public** **class** August\_05 {

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

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

//--------------------------------------------------------------------------------------------------

/\*int n,first=0,last,sum=0;

System.out.println("Enter 1 no");

n=sc.nextInt();

try{

if(n<=0){

throw new ZeroException ();

}

else{

last=n%10;

while(n>0){

first=n;

n=n/10;

}

sum=first+last;

System.out.println("Sum="+sum);

}

}

catch(ZeroException e){

System.out.println(e);

}\*/

//--------------------------------------------------------------------------------------------------

/\*int cnt=0;

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

String s1=sc.next();

try

{

for(int i=0;i<s1.length();i++)

{

char ch=s1.charAt(i);

if((ch>='a' && ch<='z')||(ch>='A' && ch<='Z'))

{

cnt++;

}

else

{

throw new NameException();

}

}

}

catch(NameException e)

{

System.out.println(e);

}

if(cnt==s1.length())

{

System.out.println("name is valid");

}\*/

//--------------------------------------------------------------------------------------------------

/\*int rollno,age,cnt=0;

String name,course;

System.out.println("Enter roll no, name, age, course");

rollno=sc.nextInt();

name=sc.next();

age=sc.nextInt();

course=sc.next();

try{

if(age>=15 && age<=21){

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

char ch=name.charAt(i);

if(Character.isLetter(ch)){

cnt++;

}

else{

throw new NameNotValidException();

}

}

if(cnt==name.length()){

student s1=new student(rollno,age,name,course);

s1.display();

}

}

else{

throw new AgeNotWithinRangeException();

}

}

catch(AgeNotWithinRangeException e){

System.out.println(e);

}

catch(NameNotValidException e1){

System.out.println(e1);

}\*/

//--------------------------------------------------------------------------------------------------

/\*Test t=new Test();

t.display();\*/

//--------------------------------------------------------------------------------------------------

/\*int m,d,y;

System.out.println("Enter day month year");

d=sc.nextInt();

m=sc.nextInt();

y=sc.nextInt();

try{

if(y>=1000 &&y<=9999){

if(m>=1&&m<=12){

if(d>=1 &&d<=31){

switch(m){

case 1:case 3:case 5:case 7:case 8:case 10:case 12:

System.out.println(d+"-"+m+"-"+y+ " Valid date");

break;

case 4:case 6:case 9:case 11:

if(d<=30)

System.out.println(d+"-"+m+"-"+y+ " Valid date");

else

System.out.println(d+"-"+m+"-"+y+ " inValid date");

break;

case 2:

if(y%4==0 && d<=29)

System.out.println(d+"-"+m+"-"+y+ " Valid date");

else if(d<=28)

System.out.println(d+"-"+m+"-"+y+ " Valid date");

else

System.out.println(d+"-"+m+"-"+y+ " inValid date");

break;

default:

break;

}

}

else{

throw new DayException();

}

}

else{

throw new MonthException();

}

}

else{

throw new YearException();

}

}

catch(YearException e){

System.out.println(e);

}

catch(MonthException e){

System.out.println(e);

}

catch(DayException e){

System.out.println(e);

}\*/

//-------------------------------------------------------------------------------------------------------------------

/\*double n1,n2;

char ch;

try {

System.out.println("Enter 1st and 2nd number ");

n1=sc.nextDouble();

n2=sc.nextDouble();

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

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

if(ch!='+' && ch!='-' && ch!='\*' && ch!='/')

throw new InvalidOperatorException();

else {

Calculator c=new Calculator(n1,n2,ch);

switch(ch) {

case '+':

c.add();

break;

case '-':

c.subtract();

break;

case '\*':

c.mul();

break;

case '/':

c.div();

break;

default:

break;

}

}

}

catch(InvalidOperatorException e) {

System.out.println(e);

}\*/

//------------------------------------------------------------------------------------------------

/\*DisplayColor d=new DisplayColor();

d.display();\*/

//--------------------------------------------------------------------------------------------------

Email e=**new** Email();

e.display();

//--------------------------------------------------------------------------------------------------

/\*try {

Patient p=new Patient();

if(p.oxyLevel<95 && p.hrtcReport>10)

throw new Positive();

else

p.display();

}

catch(Positive e) {

System.out.println(e);

}\*/

}

}