1. Write a program to read a string and validate PAN no. against following

rules: **(15 Marks)**

1. There must be eight characters.

2. First three letters must be alphabets followed by four digit number

and ends with alphabet

3. All alphabets should be in capital case.

**package** assessments;

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

**import** java.util.regex.Matcher;

**import** java.util.regex.Pattern;

**class** UserMain{

**public** **static** **boolean** testPan(String str) {

Pattern p=Pattern.*compile*("[a-zA-Z]{5}[0-9]{4}[A-Z]{1}");

Matcher m=p.matcher(str);

**boolean** b=m.matches();

**return** b;

}

}

**public** **class** PanValidation {

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

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

System.***out***.println("enter input string:");

String str=sc.nextLine();

**if**(UserMain.*testPan*(str)) {

System.***out***.println("valid pan number");

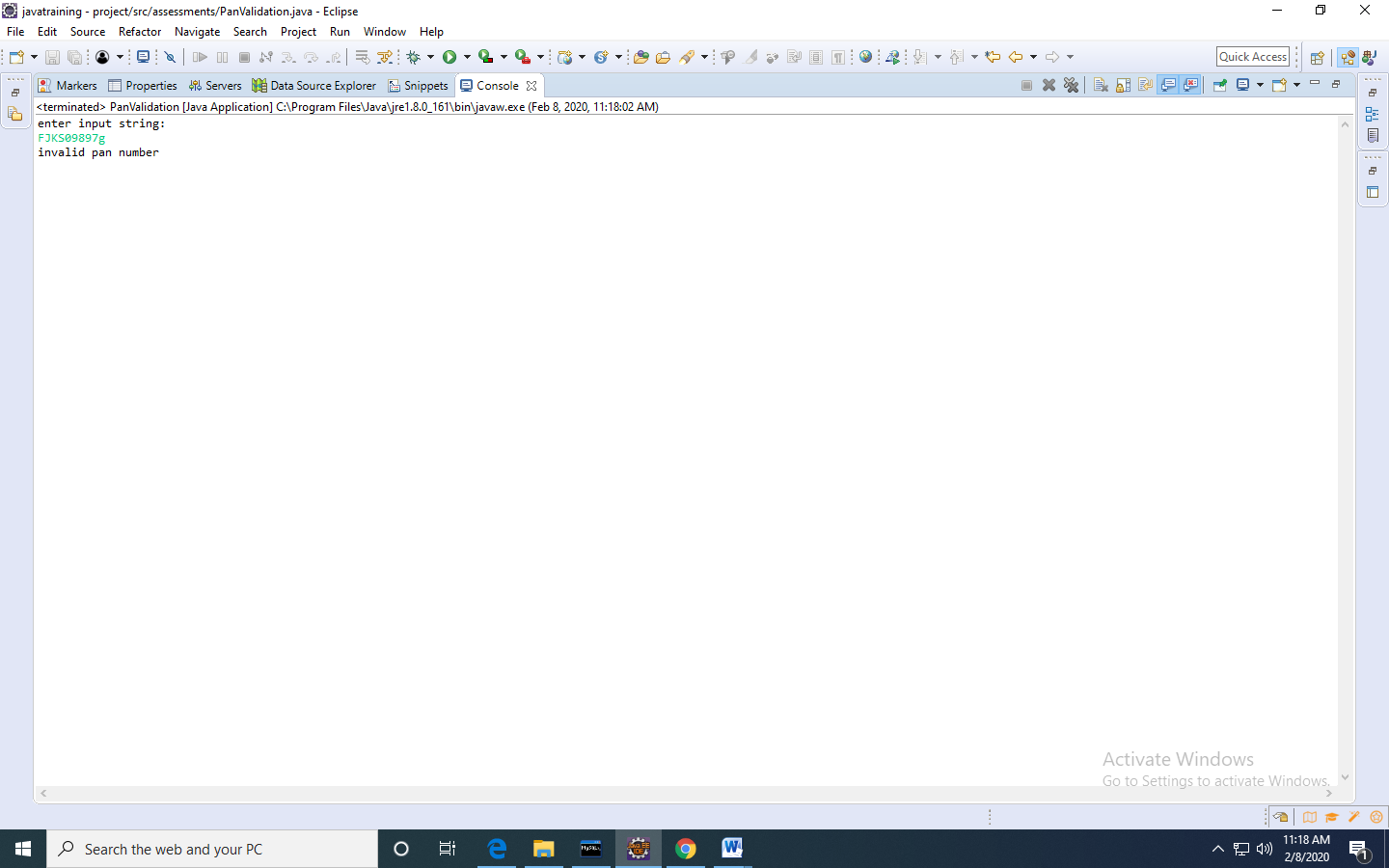
}

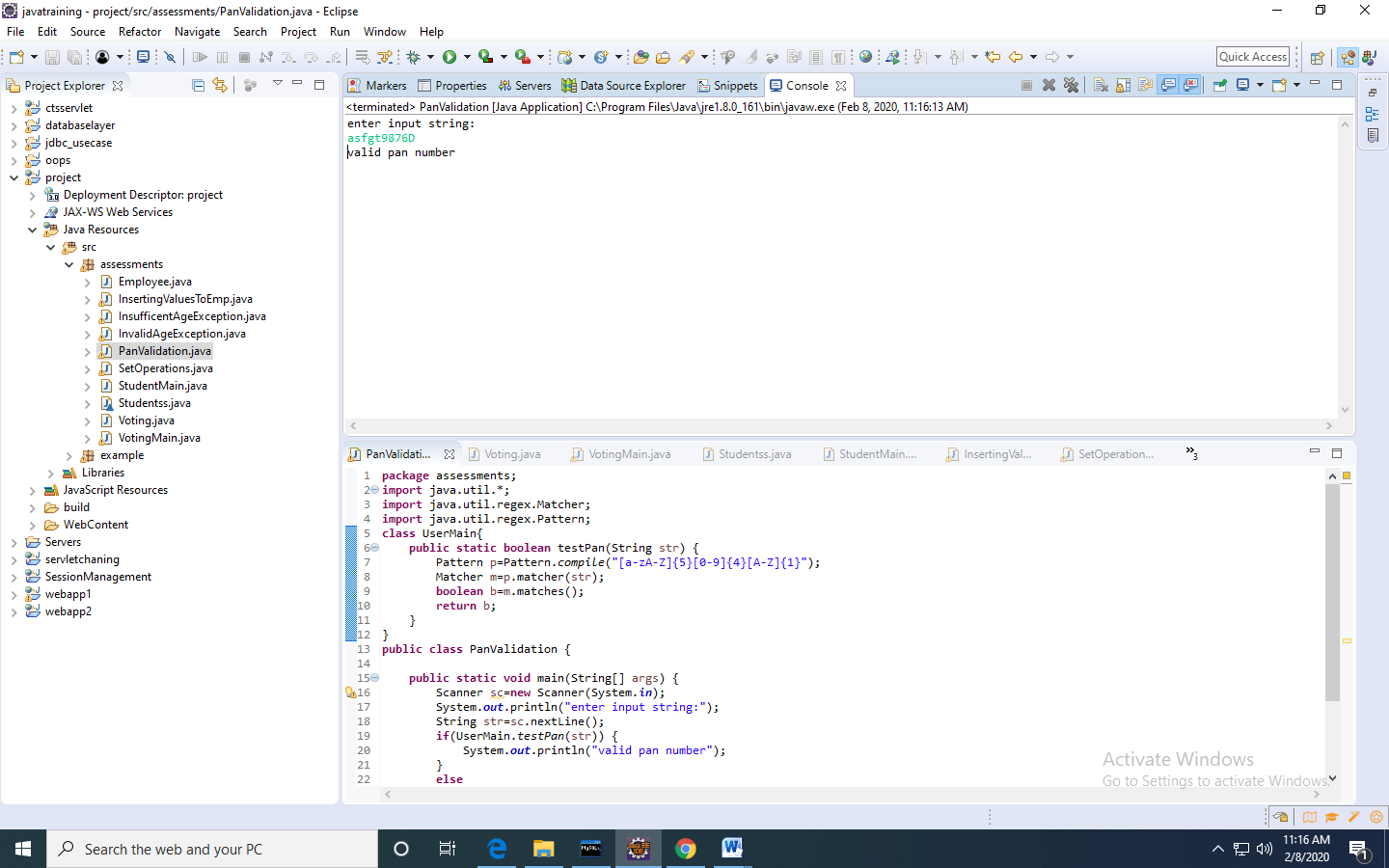
**else**

System.***out***.println("invalid pan number");

}

}





2.

Create user defined exceptions called InvalidAgeException and InsufficientAgeException, which has to be thrown when user enters a negative age and if user enters age less than 18 then it has to throw InSufficientAgeException , else has to show eligible for voting. **(15 Marks)**

InsufficentAgeException.java

**package** assessments;

**public** **class** InsufficentAgeException **extends** Exception {

InsufficentAgeException(){

**super**("insuffient age for voting");

}

}

InvalidAgeException.java

**package** assessments;

**public** **class** InvalidAgeException **extends** Exception{

InvalidAgeException(){

**super**("age value is 0 to 100 only");

}

}

Voting.java

**package** assessments;

**public** **class** Voting {

**public** **void** testAge(**int** age)**throws** InsufficentAgeException,InvalidAgeException{

**if**(age>=0 && age<18)

**throw** **new** InsufficentAgeException();

**else** **if**(age<0 || age>100)

**throw** **new** InvalidAgeException();

**else**

System.***out***.println("eligible for voting");

}

}

VotingMain.java

**package** assessments;

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

**public** **class** VotingMain {

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

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

System.***out***.println("enter age");

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

**try** {

Voting v=**new** Voting();

v.testAge(age);

}

**catch**(InsufficentAgeException e) {

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

}

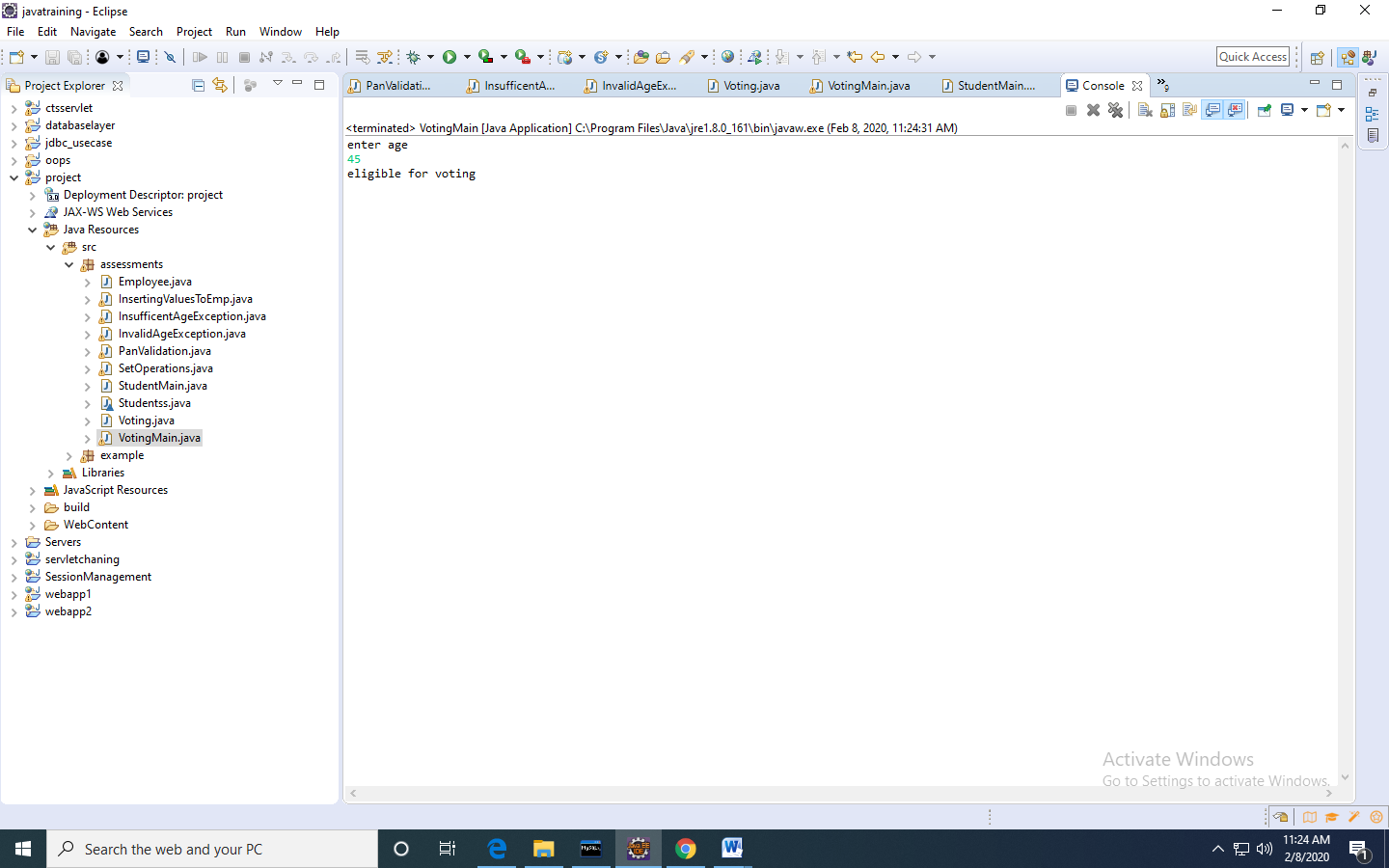
**catch**(InvalidAgeException e) {

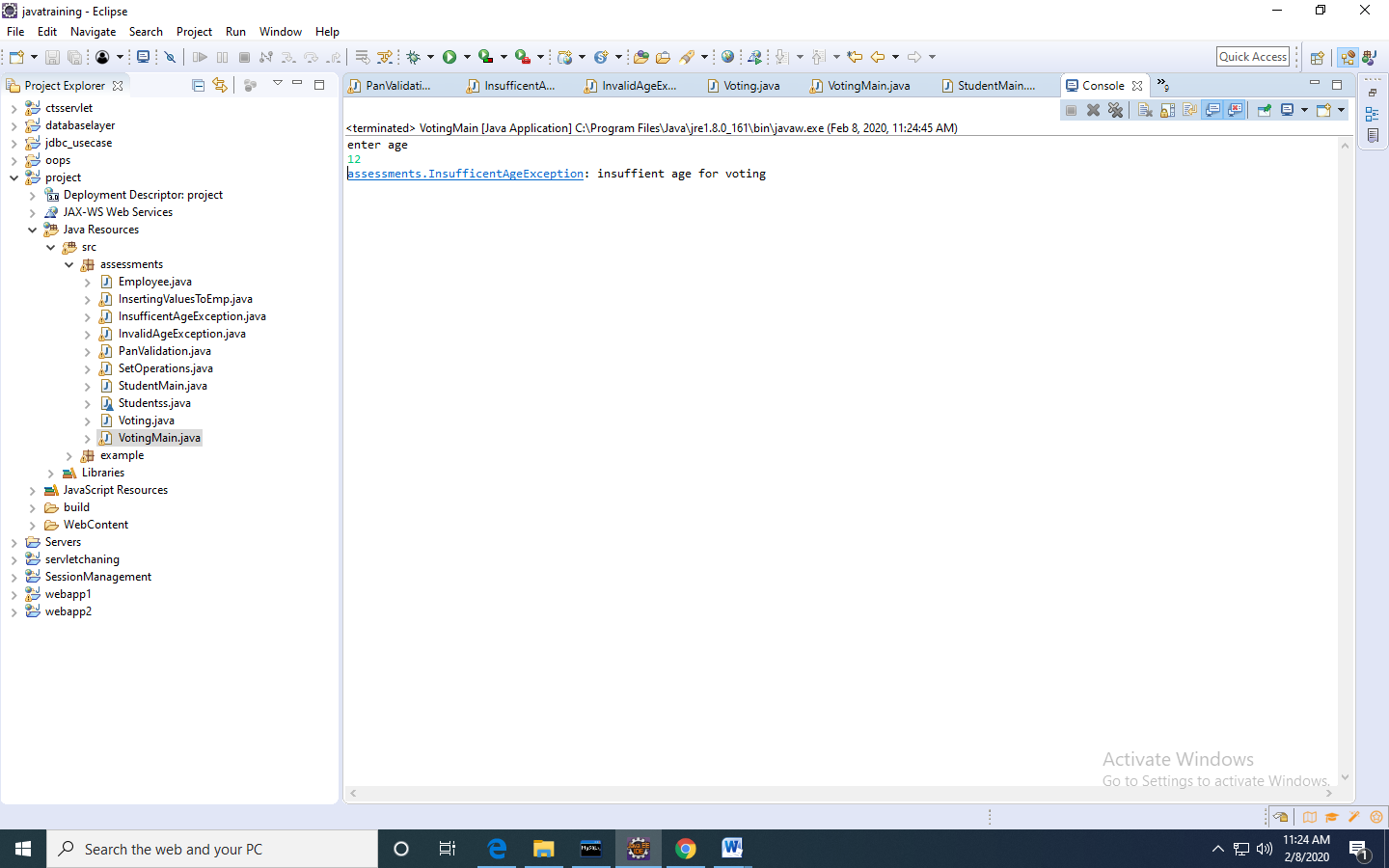
System.***out***.println(e);

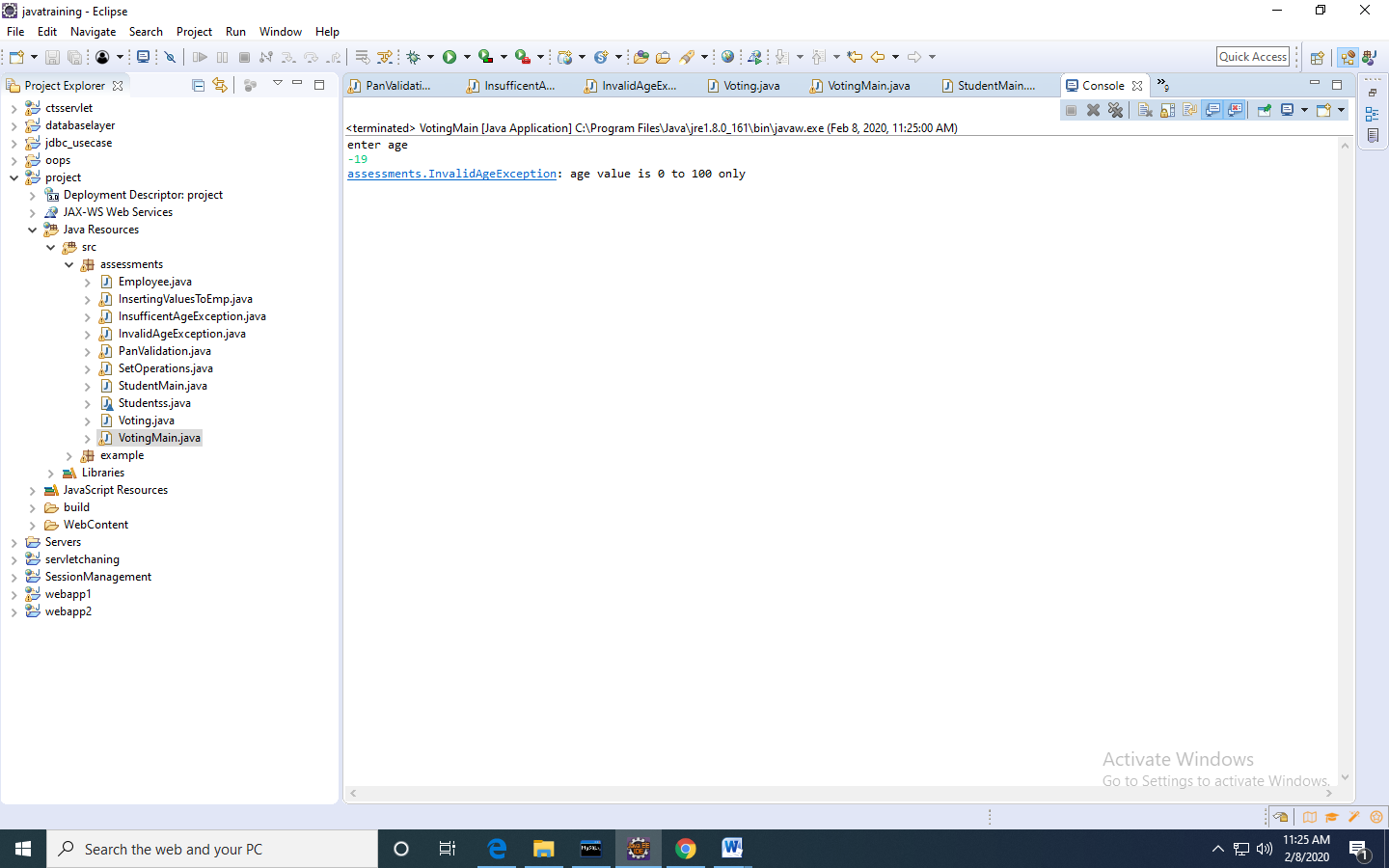
}

}

}







3.Write a java program to read student objects in a ArrayList and display the student result based on the condition that max>=80, distinction, between 60 and 79 as First class, between 50 and 50 as Second class, else fail, and display only students who are passed. **(25 Marks)**

Note; Test the pass condition by using Java8 Predicate, find the class by using Function and display the results using Consumer predefined interfaces.

Studentss.java

**package** assessments;

**class** Studentss {

String name;

**int** marks;

Studentss(String name,**int** marks){

**this**.name=name;

**this**.marks=marks;

}

}

StudentMain.java

**package** assessments;

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

**import** java.util.function.Consumer;

**import** java.util.function.Function;

**import** java.util.function.Predicate;

**public** **class** StudentMain {

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

ArrayList<Studentss>list=**new** ArrayList<Studentss>();

Studentss s1=**new** Studentss("bhavani",80);

Studentss s2=**new** Studentss("ganga",78);

Studentss s3=**new** Studentss("satish",90);

Studentss s4=**new** Studentss("krish",20);

list.add(s1);

list.add(s2);

list.add(s3);

list.add(s4);

System.***out***.println("students who are passed:");

Predicate<Studentss>p=s->s.marks>=40;

Function<Studentss,String>f=stu->{

**int** m=stu.marks;

**if**(m>=80)

**return** "Distinction";

**else** **if**(m>=60 && m<=79)

**return** "FirsT Class";

**else** **if**(m>=40 && m<=50)

**return** "Second class";

**else**

**return** "fail";

};

Consumer<Studentss>c=s->{

System.***out***.print(s.name+" ");

System.***out***.println(f.apply(s));

};

**for**(Studentss s:list) {

**if**(p.test(s))

{

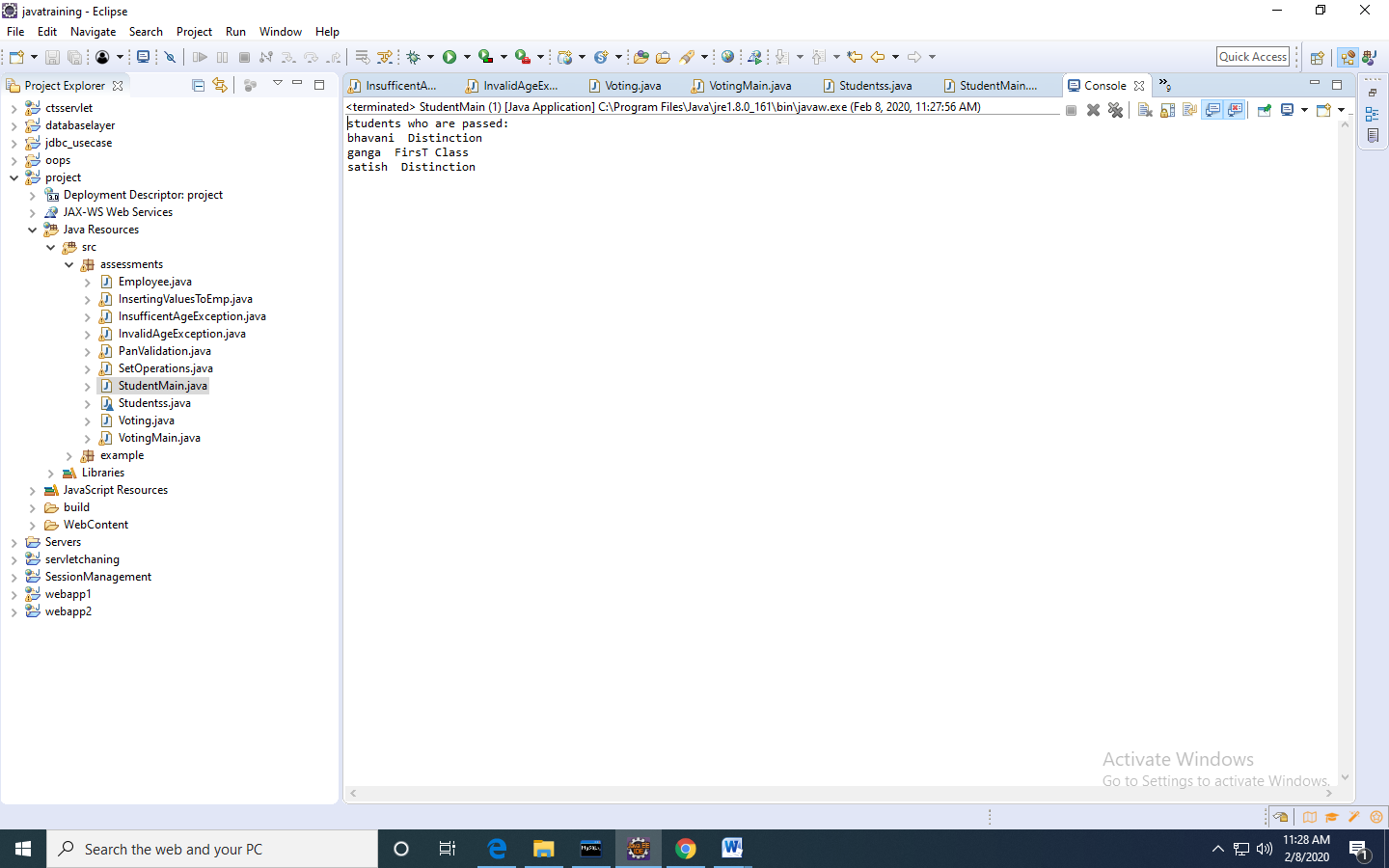
c.accept(s);

}

}

}

}



4.Write a program that performs the following actions:

1. Read 2n integers as input & a set operator (of type char).
2. Create two arraylists to store n elements in each arraylist.
3. Write a function **performSetOperations**which accepts these two arraylist and the set operator as input.
4. The function would perform the following set operations:.

'+' for SET-UNION

'\*' for SET-INTERSECTION

'-' for SET-DIFFERENCE

1. Return the resultant arraylist.

Include a class UserMainCode with the static method **performSetOperations** which accepts two arraylist and returns an arraylist.

SetOperations.java

**package** assessments;

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

**class** Operations{

**public** **static** ArrayList<Integer>setOperations(ArrayList<Integer>list1,ArrayList<Integer>list2,**char** ch){

ArrayList<Integer>list3=**new** ArrayList<Integer>();

LinkedHashSet<Integer>lh=**new** LinkedHashSet<Integer>();

**switch**(ch) {

**case** '+': list1.addAll(list2);

lh.addAll(list1);

list3.addAll(lh);

**break**;

**case** '-':

**for**(**int** i=0;i<list1.size();i++) {

**for**(**int** j=0;j<list2.size();j++) {

**if**(list1.get(i)==list2.get(j))

list1.remove(i);

}

}

list3.addAll(list1);

**break**;

**case** '\*':

**for**(**int** i=0;i<list1.size();i++) {

**for**(**int** j=0;j<list2.size();j++) {

**if**(list1.get(i)==list2.get(j))

list3.add(list1.get(i));

}

}

**break**;

}

**return** list3;

}

}

**public** **class** SetOperations {

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

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

System.***out***.println("enter size of arraylists");

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

ArrayList<Integer>list1=**new** ArrayList<Integer>();

ArrayList<Integer>list2=**new** ArrayList<Integer>();

System.***out***.println("enter first list elements");

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

list1.add(sc.nextInt());

}

System.***out***.println("enter second list elements");

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

list2.add(sc.nextInt());

}

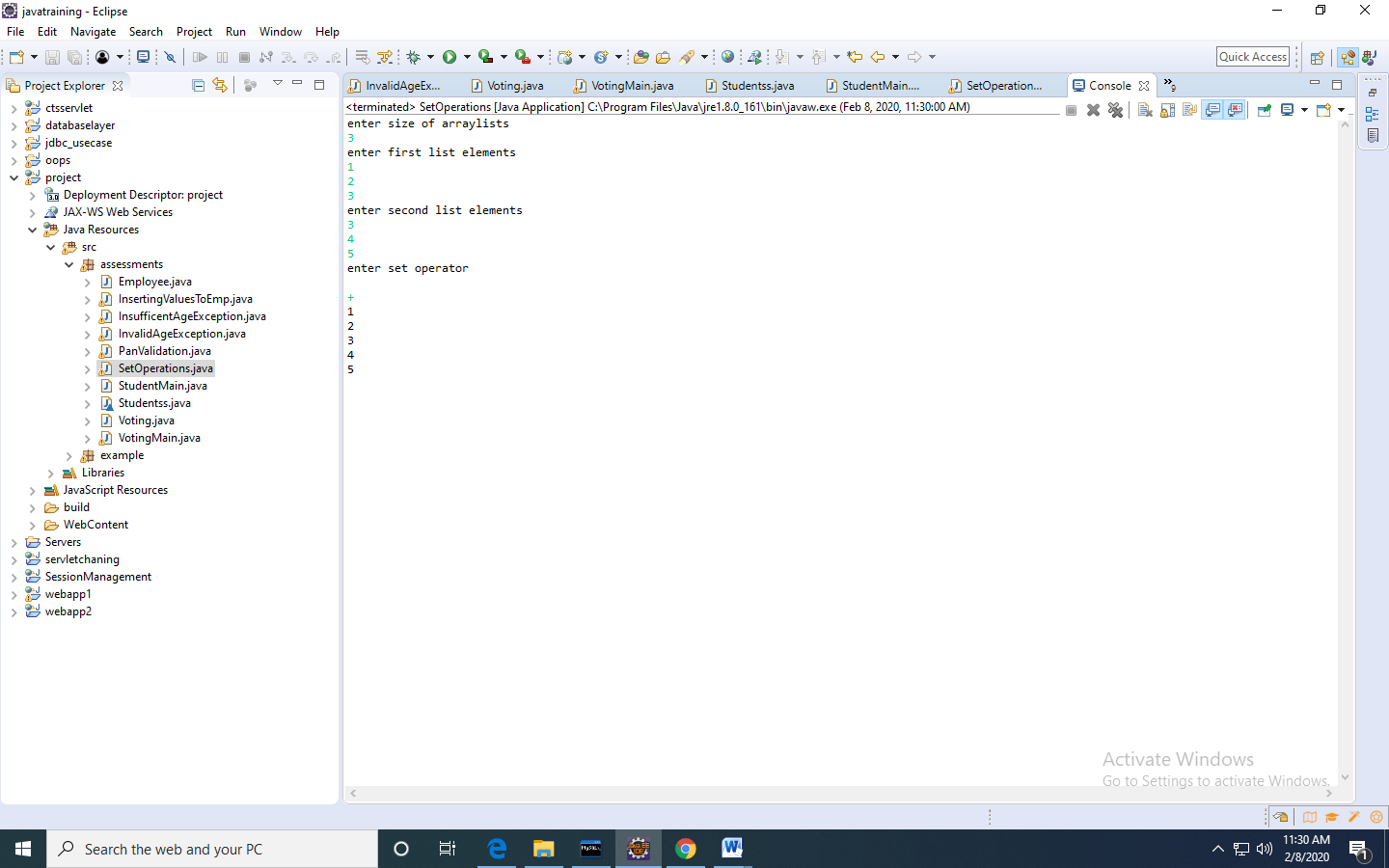
System.***out***.println("enter set operator");

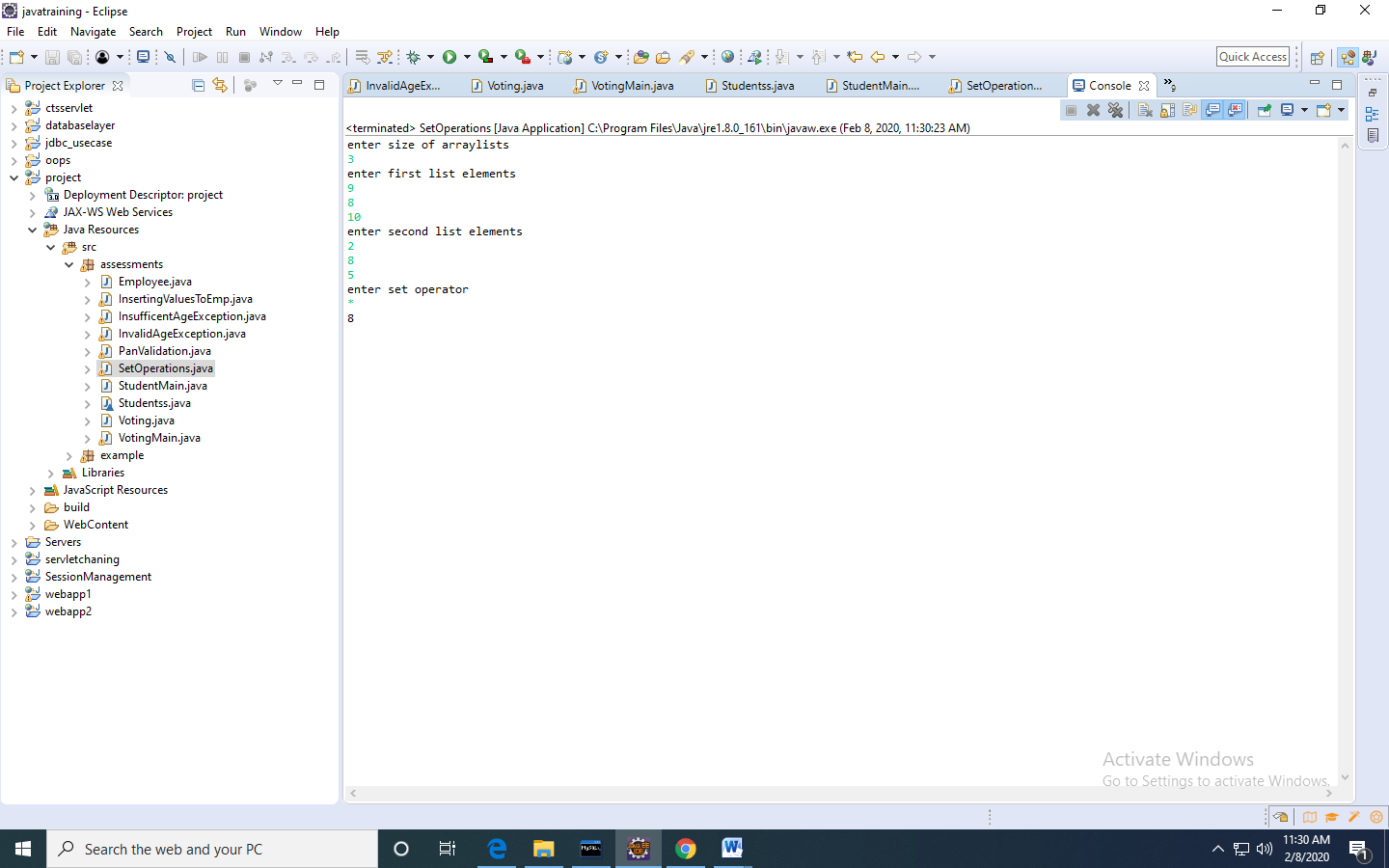
**char** ch=sc.next().charAt(0);

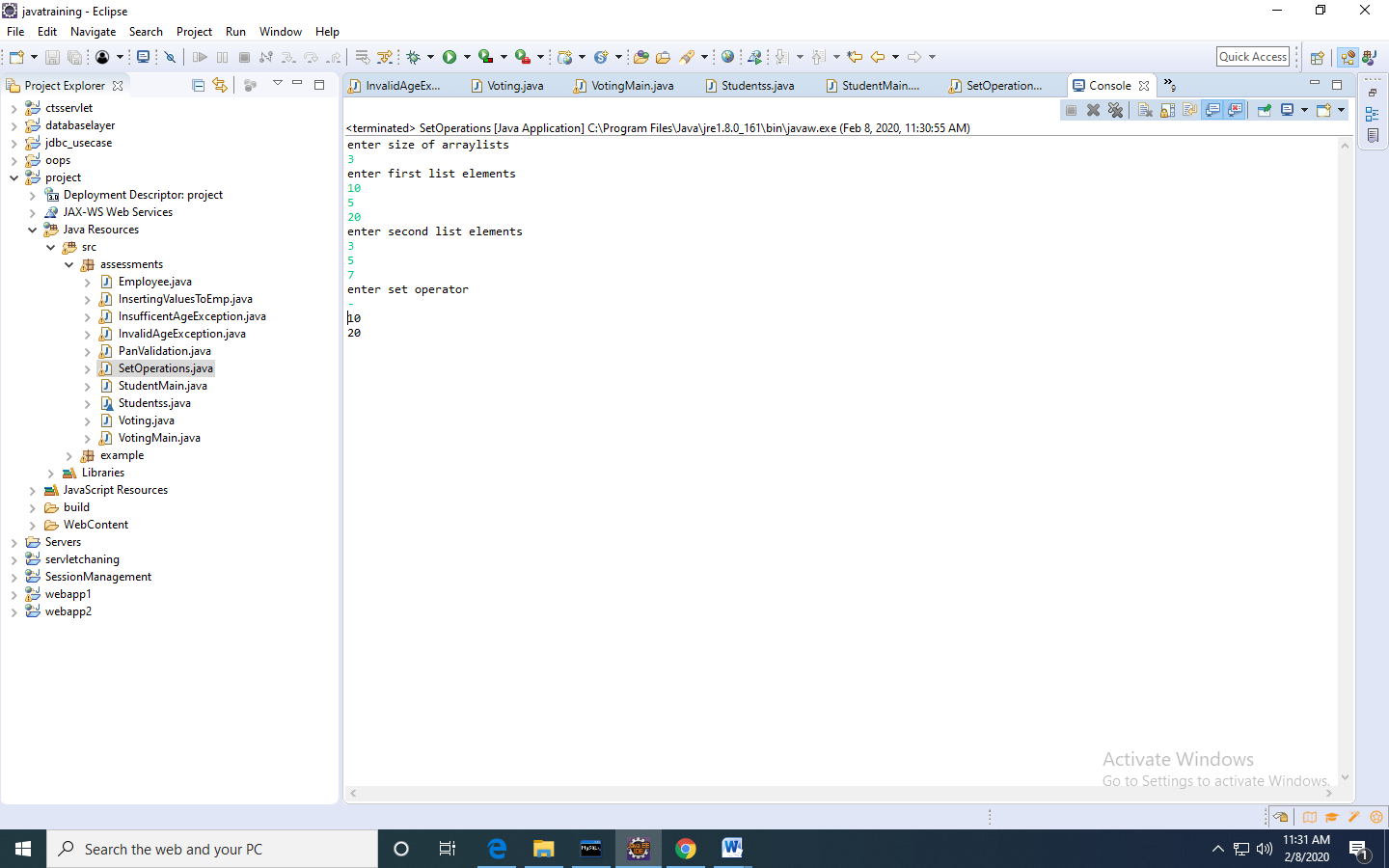
Operations.*setOperations*(list1, list2, ch).forEach(System.***out***::println);

}

}







5.Write a JDBC program to read Employee(name,id,dept,desg,company,emailed) from the user and insert into mysql table called employee.(25 Marks)

Employee.java

**package** assessments;

**public** **class** Employee {

String name;

**int** id;

String dept;

String company;

String email;

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

**public** **int** getId() {

**return** id;

}

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

**this**.id = id;

}

**public** String getDept() {

**return** dept;

}

**public** **void** setDept(String dept) {

**this**.dept = dept;

}

**public** String getCompany() {

**return** company;

}

**public** **void** setCompany(String company) {

**this**.company = company;

}

**public** String getEmail() {

**return** email;

}

**public** **void** setEmail(String email) {

**this**.email = email;

}

**public** Employee(String name, **int** id, String dept, String company, String email) {

**super**();

**this**.name = name;

**this**.id = id;

**this**.dept = dept;

**this**.company = company;

**this**.email = email;

}

@Override

**public** String toString() {

**return** "Employee [name=" + name + ", id=" + id + ", dept=" + dept + ", company=" + company + ", email=" + email

+ "]";

}

}

InsertingValuesToEmp.java

**package** assessments;

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

**import** java.sql.\*;

**public** **class** InsertingValuesToEmp {

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

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

System.***out***.println("enter name");

String name=sc.nextLine();

System.***out***.println("enter id");

**int** id=sc.nextInt();

System.***out***.println("enter department");

String dept=sc.next();

System.***out***.println("enter company name");

String company=sc.next();

System.***out***.println("enter email");

String email=sc.next();

Employee e=**new** Employee(name,id,dept,company,email);

//database connections

DriverManager.*registerDriver*( **new** com.mysql.cj.jdbc.Driver());

Connection connection=DriverManager.*getConnection*("jdbc:mysql://localhost:3306/cts","root","root");

String query="insert into employee values (?,?,?,?,?)";

PreparedStatement ps=connection.prepareStatement(query);

ps.setString(1, name);

ps.setInt(2, id);

ps.setString(3, dept);

ps.setString(4, company);

ps.setString(5, email);

**int** c=ps.executeUpdate();

**if**(c!=0)

System.***out***.println("record inserted...");

**else**

System.***out***.println("record insertion failed");

}

}

