**Ans-5**

import java.util.\*;

class Main{

public static class Citizen implements Comparable<Citizen>{

public String name;

public int age;

public String phone;

public String email;

public Citizen(String name, int age, String phone, String email)

{

this.name = name;

this.age = age;

this.phone = phone;

this.email = email;

}

public void setName(String name)

{

this.name = name;

}

public void setAge(int age)

{

this.age = age;

}

public void setPhone(String phone)

{

this.phone = phone;

}

public void setEmail(String email)

{

this.email = email;

}

public String getName()

{

return this.name;

}

public int getAge()

{

return this.age;

}

public String getPhone()

{

return this.phone;

}

public String getEmail()

{

return this.email;

}

public int compareTo(Citizen other)

{

return this.age - other.age;

}

}

public static void main(String[] args)

{

HashMap<Long, Citizen> citizensMap = new HashMap<>();

citizensMap.put((long)123457678, new Citizen("abc",40,"1234567890","abc@gmail.com"));

citizensMap.put((long)123457429, new Citizen("abcd",42,"1234567891","abcd@gmail.com"));

citizensMap.put((long)123457826, new Citizen("abce",44,"1234567892","abce@gmail.com"));

TreeMap<Long, Citizen> sortedCitizens = new TreeMap<>();

List<Citizen> citizensList = new ArrayList<>();

for(Long aadhar : citizensMap.keySet())

{

sortedCitizens.put(aadhar, citizensMap.get(aadhar));

citizensList.add(citizensMap.get(aadhar));

}

for(Long aadhar : sortedCitizens.keySet())

{

printDetails(aadhar, sortedCitizens.get(aadhar));

}

Collections.sort(citizensList);

for(Citizen c : citizensList)

{

System.out.println("Name " + c.name);

System.out.println("Age " + c.age);

System.out.println("Phone " + c.phone);

System.out.println("Email " + c.email);

}

}

public static void printDetails(Long aadhar, Citizen c)

{

System.out.println("Aadhar " + aadhar);

System.out.println("Name " + c.name);

System.out.println("Age " + c.age);

System.out.println("Phone " + c.phone);

System.out.println("Email " + c.email);

}

}

**Ans-3**

import java.util.\*;

import java.util.regex.\*;

class Main{

public static class PanCheck{

public boolean checkPan(String pan)

{

String regex = "[A-Z]{5}[0-9]{4}[A-Z]{1}";

Pattern p = Pattern.compile(regex);

Matcher m = p.matcher(pan);

return m.matches();

}

}

public static void main(String[] args)

{

Scanner sc = new Scanner(System.in);

PanCheck c1 = new PanCheck();

String pan = sc.next();

if(c1.checkPan(pan))

{

System.out.println(pan + " is valid.");

}

else

{

System.out.println(pan + " is invalid.");

}

}

}

**Ans-2**

import java.util.\*;

class Main{

public static class InvalidVotingException extends Exception{

public InvalidVotingException(String str)

{

super(str);

}

}

public static class Citizen{

private String name;

private int age;

public Citizen(String name, int age)

{

this.name = name;

this.age = age;

}

public void setName(String name)

{

this.name = name;

}

public void setAge(int age)

{

this.age = age;

}

public String getName()

{

return this.name;

}

public int getAge()

{

return this.age;

}

public String toString()

{

return this.name + " " + this.age;

}

public boolean isCitizenEligibleForVoting() throws InvalidVotingException

{

if(this.name == null || this.name.length() < 3)

{

throw new InvalidVotingException("PersonNameIsInvalid");

}

if(this.age < 18)

{

throw new InvalidVotingException("AgeIsInvalid");

}

else

{

throw new InvalidVotingException("PersonIsEligibleToVote");

}

}

}

public static void main(String[] args)

{

Citizen c1 = new Citizen("ab",12);

try{

c1.isCitizenEligibleForVoting();

}

catch(InvalidVotingException e){

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

}

}

}