21CE105 Vraj Patel

https://github.com/PatelVraj10/java-practical-file-7

1.

```java
/* By 21CE105 Vraj Patel
Question : Create a generic method for sorting an array of Comparable
objects.
GITHUB LINK : https://github.com/PatelVraj10/java-practical-file-7*/

import java.util.*;

class Student implements Comparable<Student>
{
    private String name;
    private int age;

    public Student(String name, int age)
    {
        this.name = name;
        this.age = age;
    }

    @Override
    public String toString()
    {
        return "{" + "name='" + name + '\'' + ", age=" + age + '}';
    }

    public String getName()
    {
        return name;
    }

    public int getAge()
    {
        return age;
    }

    @Override
    public int compareTo(Student o)
    {
```

21CE105 Vraj Patel

```
        if (this.age != o.getAge())
        {
            return this.age - o.getAge();
        }
        return this.name.compareTo(o.getName());
    }
}

class P7P1
{
    public static void main(String[] args)
    {
        Student[] students = { new Student("John", 15), new Student("Sam",
20),new Student("Dan", 20), new Student("Joe", 10) };

        Arrays.sort(students);
        System.out.println(Arrays.toString(students));
    }
}
```

2.

```java
/* By 21CE105 Vraj Patel
Question : Write a program that counts the occurrences of words in a text and displays the words
and their occurrences in alphabetical order of the words. Using Map and Set Classes
GITHUB LINK : https://github.com/PatelVraj10/java-practical-file-7*/

import java.util.*;

class P7P2
{
    public static void main(String... args)
    {
        Scanner sc=new Scanner(System.in);

        String str="a set of words that is complete in itself, typically containing a subject and predicate, conveying a statement, question, exclamation, or command, and consisting of a main clause and sometimes one or more subordinate clauses.";
        String[] splitStr=str.split(" ",CountSpaces(str));

        TreeMap<String,Integer> occr=new TreeMap<String,Integer>();

        for(String s:splitStr)
        {
            if(occr.containsKey(s))
            {
                int cnt=occr.get(s);
                cnt=cnt+1;
                occr.put(s,cnt);
            }
            else
            {
                occr.put(s,1);
            }
        }
        System.out.println(occr);
    }
```

21CE105 Vraj Patel

```
    public static int CountSpaces(String str)
    {
        int cnt=1;
        for(int i=0;i<str.length();i++)
        {
            if(str.charAt(i)==' ')
            {
                cnt++;
            }
        }
        return cnt;
    }
}
```

3.

```
/* By 21CE105 Vraj Patel
Question : Personal Loan Eligibility Criteria for Salaried Applicant is as
follows:
Eligible Age Group - 21 years to 60 years
Minimum Net Monthly Income - Rs. 15,000
Minimum Total Work Experience - 1 year
Citizenship - Indian
Create a class AccountHolder to store above given information entered by a
user. Create 5
objects of AccountHolder class and store them in an ArrayList. Display
names of account
holders , who are eligible to get a loan based on given criteria
GITHUB LINK : https://github.com/PatelVraj10/java-practical-file-7*/

import java.util.*;

class AccountHolder
{
    private int age;
    private float monthly_income;
    private int work_experience;
    private String citizenship;

    AccountHolder(int age,float monthly_income,int work_experience,String
citizenship)
    {
        this.age=age;
        this.monthly_income=monthly_income;
        this.work_experience=work_experience;
        this.citizenship=citizenship;
    }

    boolean isEligible()
    {
        if(this.age>=21 && this.age<=60 && this.monthly_income>=15000 &&
this.work_experience>=1 && this.citizenship.equals("Indian"))
        {
            return true;
```

```
        }
        else
        {
            return false;
        }
    }
}

class P7P3
{
    public static void main(String... args)
    {
        ArrayList<AccountHolder> arr=new ArrayList<AccountHolder>();

        arr.add(new AccountHolder(28,50000,3,"Indian"));
        arr.add(new AccountHolder(48,25000,20,"Indian"));
        arr.add(new AccountHolder(31,1000,1,"American"));
        arr.add(new AccountHolder(18,20000,0,"Indian"));
        arr.add(new AccountHolder(43,5200,5,"Pakistani"));

        for(AccountHolder A:arr)
        {
            if(A.isEligible())
            {
                System.out.println("Eligible for loan");
            }
            else
            {
                System.out.println("Not Eligible fot loan");
            }
        }
    }
}
```