

(day-13 Assignment)

1. Create an arraylist of user-defined data type Book. it should have:-

i) Name of the Book

ii) Author of the book

iii) year of publication of the book

iv) number of copies sold.

sort the array list based on the year of publication.

CODE:

```
package sba2;

import java.util.ArrayList;

class Book implements Comparable {
    String Name;
    String Author;
    Integer Year;
    int Copies;

    Book(String name, String author, int year, int copies) {
        this.Name = name;
        this.Author = author;
        this.Year = year;
        this.Copies = copies;
    }

    public int getYear() {
        return this.Year;
    }

    @Override
    public int compareTo(Object o) {
        Book b1 = (Book) o;
        return (this.Year.compareTo(b1.getYear()));
    }
}

public class Q1 {
    public static void main(String[] args) {
        ArrayList<Book> BookList = new ArrayList<Book>();
        Book b1 = new Book("War and Peace", "Leo Tolstoy", 1869,
5061570);
        Book b2 = new Book("Harry Potter and the Deathly Hallows",
"J.K.Rowling", 2007, 4475152);
        Book b3 = new Book("A Tale of Two Cities", "Charles
Dickens", 1859, 2000000);
```

```

        Book b4 = new Book("And Then There Were None", "Agatha
Christie", 1939, 1000000);
        Book b5 = new Book("The Alchemist", "Paulo Coelho", 1988,
650000);
        Book b6 = new Book("Charlotte's Web", "E.B.White", 1952,
50000);

        BookList.add(b1);
        BookList.add(b2);
        BookList.add(b3);
        BookList.add(b4);
        BookList.add(b5);
        BookList.add(b6);

        System.out.println("----- Original
Booklist -----");
        for (Book b : BookList) {
            System.out.println(b.Name + " -- " + b.Author + " -- " +
b.Year + " -- " + b.Copies);
        }
        System.out.println("-----
-----");

        Collections.sort(BookList, Collections.reverseOrder()); //
Sorted based on year (latest to oldest)

        System.out.println("----- Booklist
Sorted by year -----");
        for (Book b : BookList) {
            System.out.println(b.Name + " -- " + b.Author + " -- " +
b.Year + " -- " + b.Copies);
        }
        System.out.println("-----
-----");
    }
}

```

OUTPUT:

```
Problems Javadoc Declaration Console × Git Staging
<terminated> Q1 (5) [Java Application] C:\Users\HP\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211116-165
----- Original Booklist -----
War and Peace -- Leo Tolstoy -- 1869 -- 5061570
Harry Potter and the Deathly Hallows -- J.K.Rowling -- 2007 -- 4475152
A Tale of Two Cities -- Charles Dickens -- 1859 -- 2000000
And Then There Were None -- Agatha Christie -- 1939 -- 1000000
The Alchemist -- Paulo Coelho -- 1988 -- 650000
Charlotte's Web -- E.B.White -- 1952 -- 50000
-----
----- Booklist Sorted by year -----
Harry Potter and the Deathly Hallows -- J.K.Rowling -- 2007 -- 4475152
The Alchemist -- Paulo Coelho -- 1988 -- 650000
Charlotte's Web -- E.B.White -- 1952 -- 50000
And Then There Were None -- Agatha Christie -- 1939 -- 1000000
War and Peace -- Leo Tolstoy -- 1869 -- 5061570
A Tale of Two Cities -- Charles Dickens -- 1859 -- 2000000
-----
```

(day-14 assignment)

2. Write a program to create, write and read from a file.

CODE:

```
package sba2;

import java.io.File;

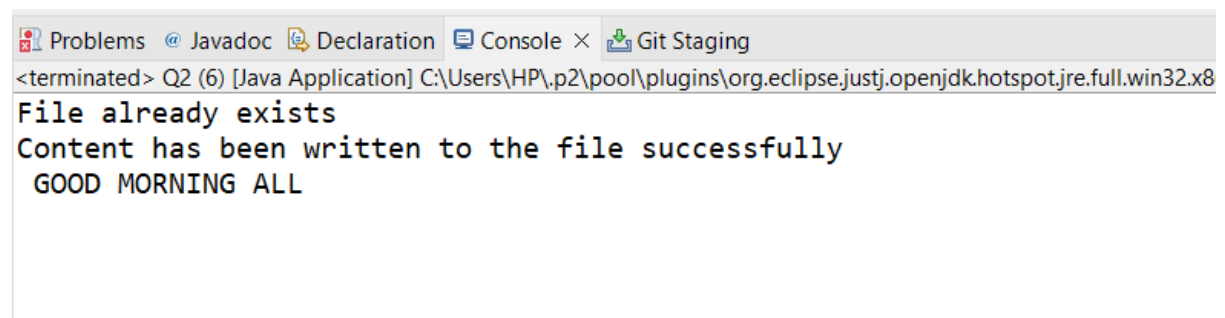
public class Q2 {
    public static void main(String[] args) {
        try
        {
            File f1=new File("E:example2.txt");
            if(f1.createNewFile())
            {
                System.out.println("a new file named
"+f1.getName()+" has been created");
            }
            else
            {
                System.out.println("File already exists");
            }
        }
        catch(IOException e)
        {
            System.out.println("an unexpected error has
occured");
            System.out.println(e);
        }
    }
    //Writing into file
}
```

```

        try
        {
            FileWriter obj1=new FileWriter("E:example2.txt");
            obj1.write(" GOOD MORNING ALL");
            obj1.close();
            System.out.println("Content has been written to the file
successfully");
        }
        catch(IOException e)
        {
            System.out.println("Some unexpected error has occured");
            System.out.println(e);
        }
    }
    //reading data
    try
    {
        File f1=new File("E:example2.txt");
        Scanner sc=new Scanner(f1);
        while(sc.hasNextLine())
        {
            String fileData=sc.nextLine();
            System.out.println(fileData);
        }
        sc.close();
    }
    catch(Exception e)
    {
        System.out.println(e);
    }
}
}

```

OUTPUT:



The screenshot shows the Eclipse IDE's Console window. The title bar includes tabs for Problems, Javadoc, Declaration, Console, and Git Staging. The console output is as follows:

```

<terminated> Q2 (6) [Java Application] C:\Users\HP\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x8
File already exists
Content has been written to the file successfully
GOOD MORNING ALL

```

3. Write a program to get the information about the file.

CODE:

```
package sba2;

import java.io.File;
public class Q3 {
    public static void main(String[] args) throws
FileNotFoundException {
        File f1=new File("E:example2.txt");

        if(f1.exists())
        {
            System.out.println("The file name is:
"+f1.getName());
            System.out.println("Is the file Writeable:
"+f1.canWrite());
            System.out.println("The absolute Path of the file
is: "+f1.getAbsolutePath());
            System.out.println("Is the file Readable:
"+f1.canRead());
            System.out.println("The size of the file in bytes:
"+f1.length());
            //reading data
            Scanner sc=new Scanner(f1);
            while(sc.hasNextLine())
            {
                String fileData=sc.nextLine();
                System.out.println(fileData);
            }
            sc.close();
        }
        else
        {
            System.out.println("the file does not exist");
        }
    }
}
```

## OUTPUT:

```
Problems @ Javadoc Declaration Console × Git Staging
<terminated> Q3 (4) [Java Application] C:\Users\HP\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.1.v20211
The file name is: example2.txt
Is the file Writeable: true
The absolute Path of the file is: E:\Java Workspace\JavaTraining\example2.txt
Is the file Readable: true
The size of the file in bytes: 17
GOOD MORNING ALL
```

4. Write a program Implement the filereader until the file ending character is “-1” and print all the data of the file.

## CODE:

```
package sba2;

import java.io.FileReader;
import java.io.IOException;

public class Q4 {
    public static void main(String[] args) {
        try
        {
            int i;
            FileReader fr= new FileReader("E:example2.txt");
            while((i=fr.read())!=-1)
            {
                System.out.print((char)i);
            }
            fr.close();
        } catch(IOException e)
        {
            System.out.println(e);
        }
    }
}
```

## OUTPUT:

