Assignment 4

Name: Priyanshu Tripathi

Roll number: 12022005

Subject: Java Programming

Code: CSPC-23

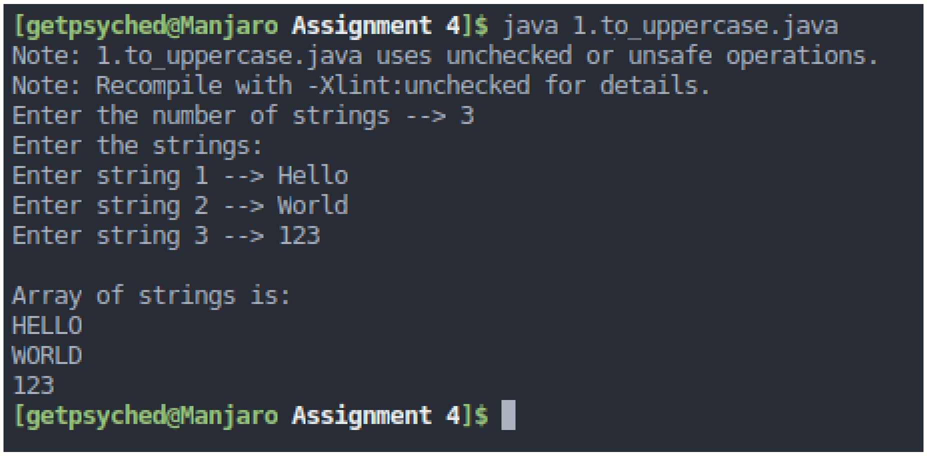
INDEX

1. to_uppercase.java

In an array list of strings, make each string uppercase. Do this with the following:

An iterator
A loop over the index values
Using the ***replaceAll*** method

```
import java.util.ArrayList;
import java.util.ListIterator;
import java.util.Scanner;

public class foobar {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        ArrayList<String> strings = new ArrayList<String>();

        System.out.print("Enter the number of strings --> ");
        int n = sc.nextInt();
        sc.nextLine();

        System.out.println("Enter the strings:");
        int i;
        for (i = 0; i < n; i++) {
            System.out.print("Enter string " + (i + 1) + " --> ");
            strings.add(sc.nextLine());
        }

        ArrayList<String> upperCaseStrings;
        upperCaseStrings = usingIterator(strings);
        upperCaseStrings = usingLoop(strings);
        upperCaseStrings = usingReplaceAll(strings);

        System.out.println("\nArray of strings is:");
        for (String string : upperCaseStrings)
            System.out.println(string);
    }

    public static ArrayList usingIterator(ArrayList<String> strings) {
        ListIterator<String> iter = strings.listIterator();
        while (iter.hasNext())
            iter.set(iter.next().toUpperCase());
        return strings;
    }

    public static ArrayList usingLoop(ArrayList<String> strings) {
        for (int i = 0; i < strings.size(); i++)
            strings.set(i, strings.get(i).toUpperCase());
        return strings;
    }

    public static ArrayList usingReplaceAll(ArrayList<String> strings) {
        ListIterator<String> iter = strings.listIterator();
        while (iter.hasNext()) {
            String temp = iter.next();
            iter.set(temp.replaceAll(temp, temp.toUpperCase()));
        }
        return strings;
    }

}
```



```
[getpsyched@Manjaro Assignment 4]$ java 1.to_uppercase.java
Note: 1.to_uppercase.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.
Enter the number of strings --> 3
Enter the strings:
Enter string 1 --> Hello
Enter string 2 --> World
Enter string 3 --> 123

Array of strings is:
HELLO
WORLD
123
[getpsyched@Manjaro Assignment 4]$
```

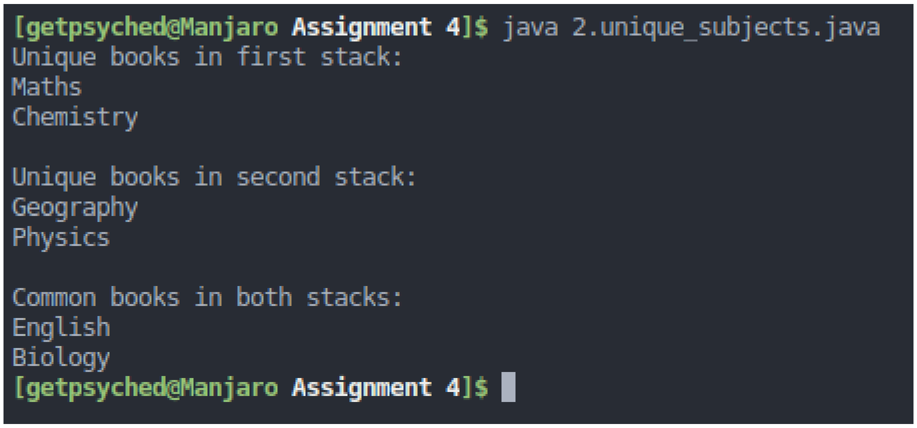
2. unique_subjects.java

Given two stacks of textbooks of the following subjects **"Chemistry"**, **"Mathematics"**, **"Biology"**, **"English"** and **"Biology"**, **"English"**, **"Geography"**, **"Physics"** respectively; find the subjects that are:

Only present in the first stack
Only present in the second stack
Present in both stacks.

(You may clone the sets to preserve the original sets from being changed by set methods.)

```
import java.util.HashSet;
import java.util.Iterator;
import java.util.Scanner;
import java.util.Set;

public class foobar {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        Set<String> set1, set2;
        set1 = Set.of("Chemistry", "Maths", "Biology", "English");
        set2 = Set.of("Biology", "English", "Geography", "Physics");
        HashSet<String> tempSet = new HashSet<String>();

        firstStack(set1, set2, tempSet);
        tempSet.clear();
        secondStack(set1, set2, tempSet);
        tempSet.clear();
        bothStacks(set1, set2, tempSet);

        sc.close();
    }

    public static void firstStack(Set<String> set1, Set<String> set2, HashSet<String> tempSet) {
        Iterator<String> iter1 = set1.iterator();
        Iterator<String> iter2 = set2.iterator();
        boolean flag = true;

        while (iter1.hasNext()) {
            iter2 = set2.iterator();
            String temp = iter1.next();
            flag = true;
            while (iter2.hasNext()) {
                if (temp == iter2.next()) {
                    flag = false;
                    break;
                }
            }

            if (flag)
                tempSet.add(temp);
        }

        System.out.println("Unique books in first stack:");
        for (String string : tempSet)
            System.out.println(string);
    }

    public static void secondStack(Set<String> set1, Set<String> set2, HashSet<String> tempSet) {
        Iterator<String> iter1 = set1.iterator();
        Iterator<String> iter2 = set2.iterator();
        boolean flag = true;
        while (iter2.hasNext()) {
            iter1 = set1.iterator();
            String temp = iter2.next();
            flag = true;
            while (iter1.hasNext()) {
                if (temp == iter1.next()) {
                    flag = false;
                    break;
                }
            }

            if (flag) {
                tempSet.add(temp);
            }
        }

        System.out.println("\nUnique books in second stack:");
        for (String string : tempSet)
            System.out.println(string);
    }

    public static void bothStacks(Set<String> set1, Set<String> set2, HashSet<String> tempSet) {
        Iterator<String> iter1 = set1.iterator();
        Iterator<String> iter2 = set2.iterator();
        boolean flag = true;

        while (iter2.hasNext()) {
            iter1 = set1.iterator();
            String temp = iter2.next();
            flag = false;

            while (iter1.hasNext()) {
                if (temp == iter1.next()) {
                    flag = true;
                    break;
                }
            }

            if (flag)
                tempSet.add(temp);
        }

        System.out.println("\nCommon books in both stacks:");
        for (String string : tempSet)
            System.out.println(string);
    }
}
```

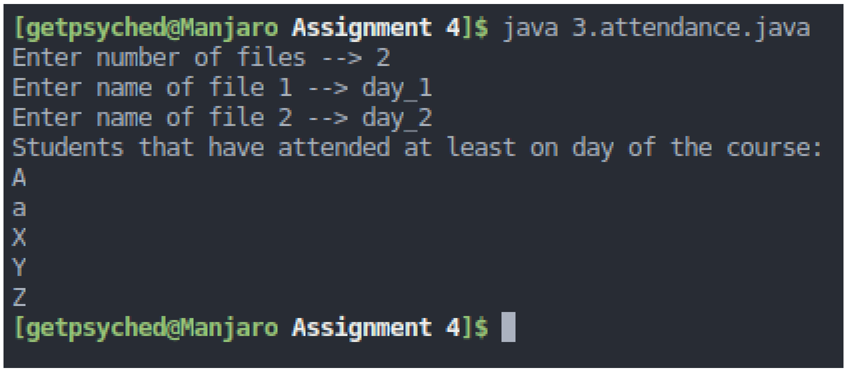


```
[getpsyched@Manjaro Assignment 4]$ java 2.unique_subjects.java
Unique books in first stack:
Maths
Chemistry

Unique books in second stack:
Geography
Physics

Common books in both stacks:
English
Biology
[getpsyched@Manjaro Assignment 4]$
```

3. attendance.java

Given one or more text files, each representing a day’s attendance in a course and containing the names of the students who attended the course on that particular day, write a program that displays, in ascending order, the names of those students who have attended at least one day of the course. The text file(s) is/are passed as command-line argument(s).

```java
import java.io.File;
import java.io.FileNotFoundException;
import java.util.HashSet;
import java.util.Scanner;
import java.util.Set;

public class foobar {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        Scanner text = new Scanner(System.in);
        Set<String> names = new HashSet<String>();

        System.out.print("Enter number of files --> ");
        int n = sc.nextInt();
        sc.nextLine();

        for (int i = 0; i < n; i++) {
            System.out.print("Enter name of file " + (i + 1) + " --> ");
            sc = new Scanner(System.in);
            File textFile = new File(sc.nextLine() + ".txt");

            try {
                text = new Scanner(textFile);
            } catch (FileNotFoundException error) {
                System.out.println("File doesn't exist.");
                return;
            }

            while (text.hasNextLine()) {
                String name = text.nextLine();
                names.add(name);
            }
        }
        System.out.println("Students that have attended at least on day of the course:");
        for (String name : names) {
            System.out.println(name);
        }
    }
}
```

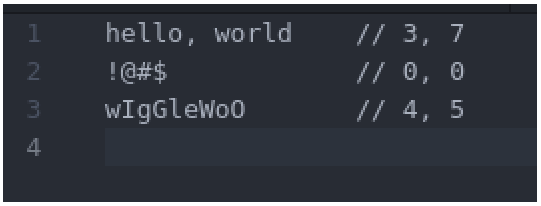


```
[getpsyched@Manjaro Assignment 4]$ java 3.attendance.java
Enter number of files --> 2
Enter name of file 1 --> day_1
Enter name of file 2 --> day_2
Students that have attended at least on day of the course:
A
a
X
Y
Z
[getpsyched@Manjaro Assignment 4]$
```

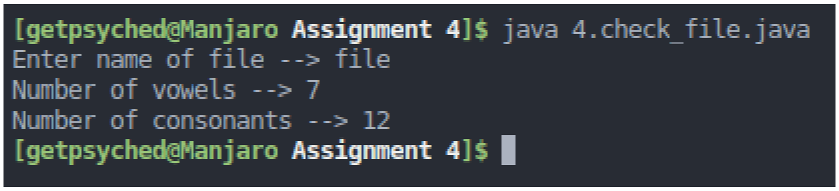
4. check_file.java

Write a program that prompts the user to enter a text file name and displays the number of vowels and consonants in the file. Use a set to store the vowels A, E, I, O, and U.

```
import java.io.File;
import java.io.FileNotFoundException;
import java.lang.Character;
import java.util.Scanner;
import java.util.Set;

public class foobar {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int vowelCounter = 0, consonantCounter = 0;
        Set<Character> vowels = Set.of('A', 'a', 'E', 'e', 'I', 'i', 'O', 'o', 'U', 'u');

        System.out.print("Enter name of file --> ");
        File textFile = new File(sc.nextLine() + ".txt");
        sc.close();

        try {
            sc = new Scanner(textFile);
        } catch (FileNotFoundException error) {
            System.out.println("File doesn't exist.");
            return;
        }

        while (sc.hasNextLine()) {
            String str = sc.nextLine();
            char arr[] = str.toCharArray();

            for (char c : arr)
                if (Character.isAlphabetic(c))
                    if (vowels.contains(c))
                        vowelCounter++;
                    else
                        consonantCounter++;
        }
        System.out.println("Number of vowels --> " + vowelCounter);
        System.out.println("Number of consonants --> " + consonantCounter);
    }
}
```



```
hello, world    // 3, 7
!@#$            // 0, 0
wIgGleWoO       // 4, 5

```



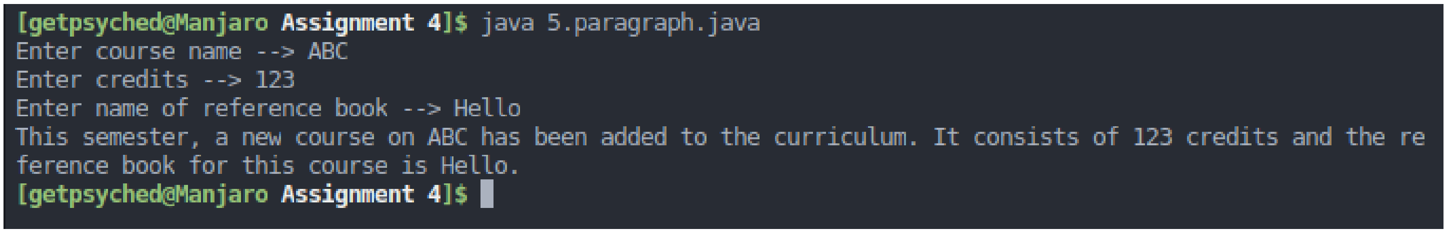
```
[getpsyched@Manjaro Assignment 4]$ java 4.check_file.java
Enter name of file --> file
Number of vowels --> 7
Number of consonants --> 12
[getpsyched@Manjaro Assignment 4]$
```

5. paragraph.java

Write a program that prompts for and reads a course name, its credits and reference book. Then print the following paragraph, inserting the appropriate data:

This semester, a new course on ***course_name*** has been added to the curriculum. It consists of ***credits*** credits and the reference book for this course is ***reference_book***.

```java
import java.util.Scanner;

public class foobar {
    public static void main(String[] args) {
        String paragraph = String.join(" ", "This semester, a new course on %s",
                "has been added to the curriculum. It consists of %s credits",
                "and the reference book for this course is %s.");
        Scanner sc = new Scanner(System.in);
        String courseName, credits, refBook;

        System.out.print("Enter course name --> ");
        courseName = sc.nextLine();
        System.out.print("Enter credits --> ");
        credits = sc.nextLine();
        System.out.print("Enter name of reference book --> ");
        refBook = sc.nextLine();

        System.out.println(String.format(paragraph, courseName, credits, refBook));

        sc.close();
    }
}
```



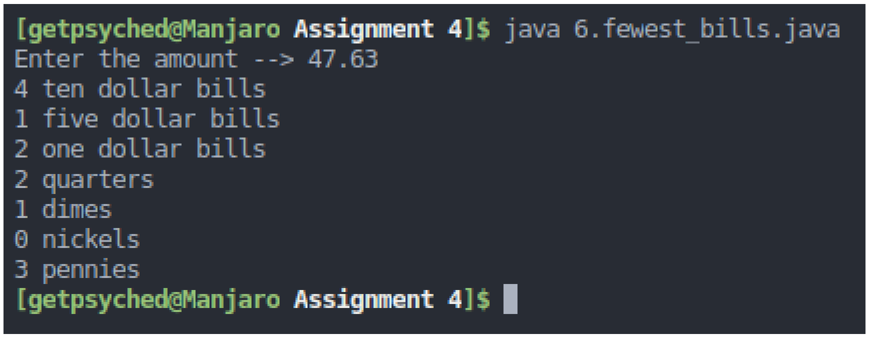
```
[getpsyched@Manjaro Assignment 4]$ java 5.paragraph.java
Enter course name --> ABC
Enter credits --> 123
Enter name of reference book --> Hello
This semester, a new course on ABC has been added to the curriculum. It consists of 123 credits and the re
ference book for this course is Hello.
[getpsyched@Manjaro Assignment 4]$ 
```

6. fewest_bills.java

Write a program that prompts for and reads a double value representing a monetary amount. Then determine the fewest number of each bill and coin needed to represent that amount, starting with the highest (assume that a ten-dollar bill is the maximum size needed). For example, if the value entered is 47.63 (forty seven dollars and sixty-three cents), then the program should print the equivalent amount as:

```
4 ten dollar bills
1 five dollar bills
2 one dollar bills
2 quarters
1 dimes
0 nickles
3 pennies
```

```
import java.util.Scanner;

public class foobar {
    public static void main(String[] args) {
        double amount;
        int one, five, ten;
        int penny, nickel, dime, quarter;

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter the amount --> ");
        amount = sc.nextDouble();

        ten = (int) amount / 10;
        amount = amount % 10;

        five = (int) amount / 5;
        amount = amount % 5;

        one = (int) amount;
        amount = amount % 1;

        quarter = (int) (amount / 0.25);
        amount = amount % 0.25;

        dime = (int) (amount / 0.10);
        amount = amount % 0.10;

        nickel = (int) (amount / 0.5);
        amount = amount % 0.5;

        penny = (int) (amount / 0.01);

        System.out.println(ten + " ten dollar bills");
        System.out.println(five + " five dollar bills");
        System.out.println(one + " one dollar bills");
        System.out.println(quarter + " quarters");
        System.out.println(dime + " dimes");
        System.out.println(nickel + " nickels");
        System.out.println(penny + " pennies");

        sc.close();
    }
}
```

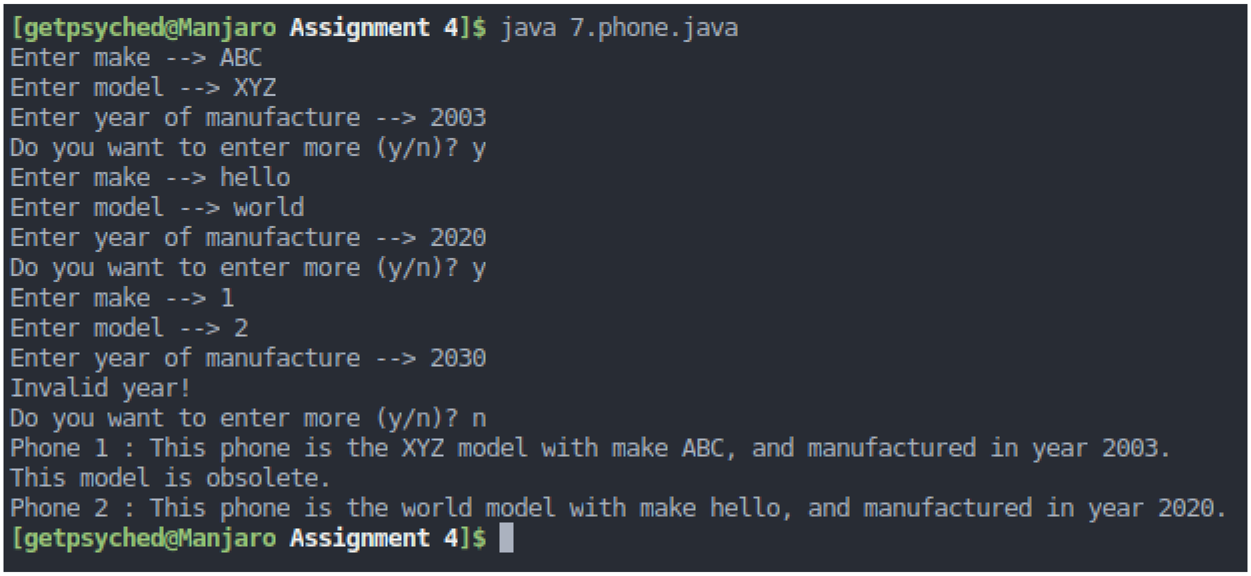


```
[getpsyched@Manjaro Assignment 4]$ java 6.fewest_bills.java
Enter the amount --> 47.63
4 ten dollar bills
1 five dollar bills
2 one dollar bills
2 quarters
1 dimes
0 nickels
3 pennies
[getpsyched@Manjaro Assignment 4]$ 
```

7. Remove duplicates from an array

Write a class called ***Phone*** that contains instance data that represents the make, model, and year of the phone. Define the ***Phone*** constructor to initialize these values. Include getter and setter methods for all instance data, and a ***toString*** method that returns a one-line description of the phone. Add a method called ***isObsolete*** that returns a boolean indicating if the phone is obsolete (if it is more than 10 years old). Create a driver class called ***PhoneCheck***, whose main method instantiates and updates several Phone objects.

```
import java.time.Year;
import java.util.ArrayList;
import java.util.Scanner;

public class PhoneCheck {
    public static void main(String[] args) {
        ArrayList<Phone> phones = new ArrayList<Phone>();
        Scanner sc = new Scanner(System.in);
        char choice;

        do {
            Phone temp = new Phone();

            System.out.print("Enter make --> ");
            temp.setMake(sc.nextLine());
            System.out.print("Enter model --> ");
            temp.setModel(sc.nextLine());
            System.out.print("Enter year of manufacture --> ");
            if (!temp.setYear(sc.nextInt()))
                System.out.println("Invalid year!");
            else
                phones.add(temp);
            sc.nextLine();

            System.out.print("Do you want to enter more (y/n)? ");
            choice = sc.nextLine().toLowerCase().charAt(0);
        } while (choice == 'y');

        int i = 1;
        for (Phone phone : phones) {
            System.out.println("Phone " + i + " : " + phone.phoneToString());
            if (phone.isObsolete())
                System.out.println("This model is obsolete.");
            ++i;
        }

        sc.close();
    }
}

class Phone {

    private String make, model;
    private int year;
    private static int currentYear = Year.now().getValue();

    Phone() {
        make = null;
        model = null;
        year = 0;
    }
    String getMake() {
        return make;
    }

    void setMake(String customMake) {
        make = customMake;
    }

    String getModel() {
        return model;
    }

    void setModel(String customModel) {
        model = customModel;
    }

    int getYear() {
        return year;
    }

    boolean setYear(int customYear) {
        if (customYear > currentYear)
            return false;
        else
            year = customYear;
        return true;
    }

    String phoneToString() {
        String formatString = "This phone is the %s model with make %s, and manufactured in year
%d.";

        return String.format(formatString, model, make, year);
    }

    boolean isObsolete() {
        return (currentYear - year >= 10);
    }
}
```



```
[getpsyched@Manjaro Assignment 4]$ java 7.phone.java
Enter make --> ABC
Enter model --> XYZ
Enter year of manufacture --> 2003
Do you want to enter more (y/n)? y
Enter make --> hello
Enter model --> world
Enter year of manufacture --> 2020
Do you want to enter more (y/n)? y
Enter make --> 1
Enter model --> 2
Enter year of manufacture --> 2030
Invalid year!
Do you want to enter more (y/n)? n
Phone 1 : This phone is the XYZ model with make ABC, and manufactured in year 2003.
This model is obsolete.
Phone 2 : This phone is the world model with make hello, and manufactured in year 2020.
[getpsyched@Manjaro Assignment 4]$
```