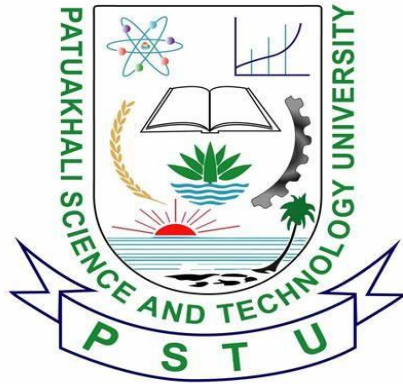# PATUAKHALI SCIENCE AND TECHNOLOGY UNIVERSITY

Course Code: CCE-121

Assignment – 10

SUBMITTED TO:

Prof. Dr. Md Samsuzzaman

Department of Computer and Communication Engineering
Faculty of Computer Science and Engineering

SUBMITTED BY

Name: **Tanmoy Kumar Das**

ID: **2102079**, Registration No: **10206**

Faculty of Computer Science and Engineering

Date of submission: 20-12-2023

Exercise:

14.1 a) False. String objects are compared using operator == to determine whether they're the

 same object in memory.

 b) False. String objects are immutable and cannot be modified after they're created.

 StringBuilder objects can be modified after they're created.

 14.2: a) s1.equals(s2)

 b) s1 += s2;

c) s1.length()

14.3:) A palindrome is a word that reads the same both forward and backward, such as 'radar'and 'madam'. Write an application to check if a string entered by the user is a palindrome or not.

Ans: **import java.util.Scanner;**

public class palindrome {

  public static void main(String[] args) {

    Scanner input=new Scanner(System.in);

    String str=input.next();

    String org_str=str;

    String rev="";

    int len=str.length();

    for(int i=len-1;i>=0;i--)

    {

  rev=rev+str.charAt(i);

    }

```java
        if(org_str.equals(rev))

        {

            System.out.println(org_str+" this is palindrome");

        }

        else

        {

            System.out.println(org_str+" Is not palindrome");

        }

    }

}
```

14.4 (Comparing Portions of Strings) Write an application that uses String method region

```java
import java.util.Scanner;

public class compareRegion {

    public static void main(String[] args) {

        Scanner input=new Scanner(System.in);

        String first=input.nextLine();

        String second=input.nextLine();

        int numChar=input.nextInt();

        int startIndex=input.nextInt();

if(startIndex>=0&&startIndex<first.length()&&startIndex<second.length()&&first.regionMatches(true,startIndex, second, numChar, startIndex))

        {

         System.out.println("The specified portion of the string are equal");

        }

        else{

            System.out.println("IS not equal");
```

```
        }

    }

}
```

14.5:(Random Sentences)

Ans: import java.util.Random;

```java
public class exfiveChapterfourteen{
    public static void main(String[] args) {
        String []articles={"the","a","one","some","any"};
        String [] nouns={"boy","girl","dog","town","car"};
        String [] verbs={"dropped","ran","jumped","skipped","walked"};
        String [] prepositions={"to","from","over","under","on"};
        Random random=new Random();
        for(int i=0;i<20;i++)
        {
            String article=articles[random.nextInt(articles.length)];
            String noun1=nouns[random.nextInt(nouns.length)];
            String verb=verbs[random.nextInt(verbs.length)];
            String preposition=prepositions[random.nextInt(prepositions.length)];
            String article2=articles[random.nextInt(articles.length)];
            String noun2=nouns[random.nextInt(nouns.length)];
            String sentence=capitalize(article)+" "+noun1+" "+verb+" "+preposition+" "+article2+" " +noun2+".";
            System.out.println(sentence);
        }
```

```
    }
    public static String capitalize(String s)
    {
        return s.substring(0,1).toUpperCase()+s.substring(1 );
    }


}
```

14.6: (Project: Limericks)

Ans: import java.util.Random;

```
public class LimerickGenerator {
    public static void main(String[] args) {
        for (int i = 0; i < 5; i++) {
            String line1 = generateLine();
            String line2 = generateLine();
            String line3 = generateLine();
            String line4 = generateLine();
            String line5 = generateLine();
            System.out.println(line1);
            System.out.println(line2);
            System.out.println(line3);
            System.out.println(line4);
            System.out.println(line5);
            System.out.println();
        }
```

```java
    }
    private static String generateLine() {

        String[] subjects = {"a cat", "a dog", "a man", "a woman", "a frog"};

        String[] verbs = {"sat", "stood", "jumped", "ran", "sang"};

        String[] adverbs = {"merrily", "quickly", "happily", "slowly", "loudly"};

        String[] rhymes = {"fun", "sun", "bun", "run", "done"};

        Random random = new Random();

    String line = "There once was " + subjects[random.nextInt(subjects.length)] +
"\n" + "Who " + verbs[random.nextInt(verbs.length)] + " " +
adverbs[random.nextInt(adverbs.length)] + "\n"+   "It was really quite " +
rhymes[random.nextInt(rhymes.length)] + "\n";

    return line;

    }
}
```

14.7 (Pig Latin)

```java
import java.util.Scanner;

 public class PigLatin {

 public static void main(String[] args) {

 Scanner input = new Scanner(System.in);

 System.out.println("Enter a sentence: ");

}

 String sentence = input.nextLine();

 input.close();

 String[] words = sentence.split(" ");

 for (String word : words) {
```

```java
            System.out.print(word.substring(1) + word.charAt(0) + "ay ");
        }
    }
```

14.8 (Tokenizing Telephone Numbers)

```java
import java.util.Scanner;

public class TokeinizingTelephone {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);

        System.out.println("Enter a telephone number: ");

        String telephoneNumber = input.nextLine();

        input.close();

        String[] tokens = telephoneNumber.split("[()\\- ]");

        String areaCode = tokens[1];

        String firstThreeDigits = tokens[3];

        String lastFourDigits = tokens[4];

        String phoneNumber = firstThreeDigits + lastFourDigits;

        System.out.println("Area code: " + areaCode);

        System.out.println("Phone number: " + phoneNumber);

    }

19 }
```

14.9 (Displaying a Sentence with Its Words Reversed)

```java
import java.util.Scanner;

public class ReverseSentence {

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

```java
        Scanner input = new Scanner(System.in);

        System.out.println("Enter a sentence: ");

        String sentence = input.nextLine();

        input.close();

        }

        String[] words = sentence.split(" ");

        for (int i = words.length - 1; i >= 0; i--) {

        System.out.print(words[i] + " ");

        }

        }
```

14.10 (Longest Word in a Sentence)

```java
import java.util.Scanner;


public class LongestWord {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);

        System.out.println("Enter a sentence: ");

        String sentence = input.nextLine();

        input.close();

        String[] words = sentence.split(" ");

        int maxLength = 0;

        String longest_word = "";

        for (String word : words) {
```

```java
if (word.length() > maxLength) {

longest_word = word;

maxLength = word.length();

}

}

System.out.println("The longest word is: " + longest_word);

}

}
```