Implementations of String Methods
Core2Web Technology

==

1. Concate Method

```
class marvel {
    static String myConcat(String str1, String str2){
        char[] ch1 = str1.toCharArray();
        char[] ch2 = str2.toCharArray();
        int l1 = ch1.length;
        int l2 = ch2.length;
        char[] result = new char[l1+l2];
        int index=0;
        for(int i=0;i<l1;i++){
            result[index++]=ch1[i];
        }
        for(int i=0;i<l2;i++){
            result[index++]=ch2[i];
        }
        String str = new String(result);
        return str;
    }
    public static void main(String[] args){
        String str1 = "Shashi";
        String str2 = "Bagal";
        System.out.println(myConcat(str1,str2));
    }
}
```

2. Length Method

```
class marvel {
    static int myLength(String str){
        char[] arr = str.toCharArray();
        int len = 0;
        for(int i=0;i<arr.length;i++){
            len++;
        }
        return len;
    }
    public static void main(String[] args){
        String str = "Shashi";
        System.out.println(myLength(str));
    }
}
```

3. charAt Method

```
class marvel {
    static char myCharAt(String str, int index){
        char[] arr = str.toCharArray();
        return arr[index];
    }
    public static void main(String[] args){
        String str = "Shashi";
        System.out.println(myCharAt(str,4));
    }
}
```

4. compareTo Method

```
class marvel {
    static int myCompareTo(String str1, String str2){
        char[] arr1 = str1.toCharArray();
        char[] arr2 = str2.toCharArray();
        int l1 = arr1.length;
        int l2 = arr2.length;
        if(l1==l2){
            for(int i=0;i<l1;i++){
                if(arr1[i]!=arr2[i])
                    return arr1[i]-arr2[i];
            }
        }
        return l1-l2;
    }
    public static void main(String[] args){
        String str1 = "Shashi";
        String str2 = "Ashish";
        System.out.println(myCompareTo(str1,str2));
    }
}
```

5. compareToIgnoreCase Method

```
class marvel {
    static int myCompareToIgnoreCase(String str1, String str2){
        char[] arr1 = str1.toCharArray();
        char[] arr2 = str2.toCharArray();
        int l1 = arr1.length;
        int l2 = arr2.length;
        if(l1==l2){
            for(int i=0;i<l1;i++){
                if(arr1[i]!=arr2[i] && arr1[i]!=arr2[i]+32 && arr1[i]!=arr2[i]-32)
                    return arr1[i]-arr2[i];
            }
        }
        return l1-l2;
    }
    public static void main(String[] args){
        String str1 = "ShAshi";
        String str2 = "shBshi";
        System.out.println(myCompareToIgnoreCase(str1,str2));
    }
}
```

6. equals Method

```
class marvel {
    static boolean myequals(String str1, String str2){
        char[] arr1 = str1.toCharArray();
        char[] arr2 = str2.toCharArray();
        if(arr1.length!=arr2.length)
            return false;
        for(int i=0;i<arr1.length;i++){
            if(arr1[i]!=arr2[i])
                return false;
        }
        return true;
    }
    public static void main(String[] args){
        String str1 = "Shashi";
        String str2 = "Shashi";
        System.out.println(myequals(str1,str2));
    }
}
```

7. equalsIgnoreCase Method

```
class marvel {
    static boolean myequalsIgnoreCase(String str1, String str2){
        char[] arr1 = str1.toCharArray();
        char[] arr2 = str2.toCharArray();
        if(arr1.length!=arr2.length)
            return false;
        for(int i=0;i<arr1.length;i++){
            if(arr1[i]!=arr2[i] && arr1[i]!=arr2[i]+32 && arr1[i]!=arr2[i]-32)
                return false;
        }
        return true;
    }
    public static void main(String[] args){
        String str1 = "SHASHI";
        String str2 = "shashi";
        System.out.println(myequalsIgnoreCase(str1,str2));
    }
}
```

8. startsWith Method

```
class marvel {
    static boolean mystartsWith(String str, String prefix, int index){
        if(index<0 || index>=str.length() || ((str.length()-index)-
(prefix.length())<0))
            return false;

        char[] arr1 = str.toCharArray();
        char[] arr2 = prefix.toCharArray();

        for(int i=index;i<arr1.length;i++){
            int temp=i;
            int count = 0;
            for(int j=0;j<arr2.length;j++){
                if(arr1[i++]==arr2[j])
                    count++;
            }
            i=temp;
            if(count==arr2.length)
                return true;
        }
        return false;
    }
    public static void main(String[] args){
        String str = "Core2Web";
        System.out.println(str.startsWith("br",7));
    }
}
```

9. endsWith Method

```
class marvel {
    static boolean myendsWith(String str, String suffix){
        char[] arr1 = str.toCharArray();
        char[] arr2 = suffix.toCharArray();
        int check = arr1.length-arr2.length;
        for(int i=check;i<arr1.length;i++){
            int temp = i;
            int count=0;
            for(int j=0;j<arr2.length;j++){
                if(arr1[i++]==arr2[j])
                    count++;
            }
            if(count==arr2.length)
                return true;
        }
        return false;
    }
    public static void main(String[] args){
        String str = "Know the code till the core";
        System.out.println(myendsWith(str,"core"));
    }
}
```

10. indexOf Method

```
class marvel {
    static int myindexOf(String str, char ch, int index){
        char[] arr = str.toCharArray();
        if(index<0)
            index=0;
        for(int i=index;i<arr.length;i++){
            if(arr[i]==ch)
                return i;
        }
        return -1;
    }
    public static void main(String[] args){
        String str = "Shashi";
        System.out.println(myindexOf(str,'h',0));
        System.out.println(myindexOf(str,'h',1));
        System.out.println(myindexOf(str,'h',2));
        System.out.println(myindexOf(str,'h',-20));
    }
}
```

11. lastIndexOf Method

```
class marvel {
    static int mylastIndexOf(String str, char ch, int uptoIndex){
        char[] arr = str.toCharArray();
        int result=-1;
        for(int i=0;i<=uptoIndex;i++){
            if(arr[i]==ch)
                result = i;
        }
        return result;
    }
    public static void main(String[] args){
        String str = "Shashi";

        System.out.println(mylastIndexOf(str,'h',0));
        System.out.println(mylastIndexOf(str,'h',5));
        System.out.println(mylastIndexOf(str,'h',-20));

        System.out.println(str.lastIndexOf('h',0));
        System.out.println(str.lastIndexOf('h',5));
        System.out.println(str.lastIndexOf('h',-20));
    }
}
```

12. replace Method

```
class marvel {
    static String myreplace(String str, char oldChar, char newChar){
        char[] arr = str.toCharArray();
        for(int i=0;i<arr.length;i++){
            if(arr[i]==oldChar)
                arr[i]=newChar;
        }
        String result = new String(arr);
        return result;
    }
    public static void main(String[] args){
        String str = "Shashi";
        System.out.println(myreplace(str,'h','i'));
    }
}
```

13. substring(int index)

```
class marvel {
    static String mysubstring(String str, int index){
        char[] arr = str.toCharArray();
        char[] result = new char[arr.length-index];
        int ind =0;
        for(int i=index;i<arr.length;i++){
            result[ind++]=arr[i];
        }
        return new String(result);
    }
    public static void main(String[] args){
        String str = "ShashiBagal";
        System.out.println(mysubstring(str,6));
    }
}
```

14. substring(int start, int end)

```
class marvel {
    static String mysubstring(String str, int start, int end){
        char[] arr = str.toCharArray();
        char[] result = new char[end-start];
        int index=0;
        for(int i=start;i<end;i++){
            result[index++]=arr[i];
        }
        return new String(result);
    }
    public static void main(String[] args){
        String str = "ShashiCore2WebBagal";
        System.out.println(str.substring(6,14));
        System.out.println(mysubstring(str,6,14));
    }
}
```

15. toLowerCase Method

```
class marvel {
    static String mytoLowerCase(String str){
        char[] arr = str.toCharArray();
        for(int i=0;i<arr.length;i++){
            if(arr[i]<97)
                arr[i]=(char)(arr[i]+32);
        }
        return new String(arr);
    }
    public static void main(String[] args){
        String str = "ShashiBagal";
        System.out.println(str.toLowerCase());
        System.out.println(mytoLowerCase(str));
    }
}
```

16. toUpperCase Method

```
class marvel {
    static String mytoUpperCase(String str){
        char[] arr = str.toCharArray();
        for(int i=0;i<arr.length;i++){
            if(arr[i]>90)
                arr[i]=(char)(arr[i]-32);
        }
        return new String(arr);
    }
    public static void main(String[] args){
        String str = "shashibagal";
        System.out.println(str.toUpperCase());
        System.out.println(mytoUpperCase(str));
    }
}
```

17. trim Method

```
public class Main {
    static String mytrim(String str){
        char[] arr = str.toCharArray();

        // for starting blank spaces
        int start=0;
        for(int i=0;i<arr.length;i++){
            if(arr[i]==32){
                start=i;
            }
            else{
                break;
            }
        }
        start++;
        // for ending blank places
        int end=arr.length-1;
        for(int i=arr.length-1;i>=0;i--){
            if(arr[i]==32){
                end=i;
            }
            else{
                break;
            }
        }
        int size = end-start+1;
        char[] result = new char[size];
        for(int i=0;i<size;i++){
            result[i]=arr[start++];
        }
        return new String(result);
    }
    public static void main(String[] args){
        String str = "  Shashi Bagal";
        System.out.println(str);
        System.out.println(str.trim());

        System.out.println(mytrim(str));
    }
}
```

18. split Method

```
public class Main {
    static String[] mysplit(String str, String delimiter){
        char[] arr = str.toCharArray();
        char[] sample = delimiter.toCharArray();
        char ch = sample[0];
        int count=0;
        for(int i=0;i<arr.length;i++){
            if(arr[i]==ch){
                count++;
            }
        }
        int index=0;
        String[] result = new String[count+1];
        for(int i=0;i<arr.length;i++){
            result[index] = ""; // initialize each element to an empty string
            while(i < arr.length && arr[i] != ch){
                result[index] += arr[i];
                i++;
            }
            index++;
        }
        return result;
    }
    public static void main(String[] args){
        String str = "i love you";
        String[] arr = mysplit(str," ");
        for(int i=0;i<arr.length;i++){
            System.out.println(arr[i]);
        }
    }
}
```