Experiment No 1

```
const prompt=require('prompt-sync') ();
function calculateTriangleArea(base, height) {
    return 0.5 * base * height;
  }

  function calculateRectangleArea(length, width) {
    return length * width;
  }

  function calculateCircleArea(radius) {
    return Math.PI * radius * radius;
  }

  const triangleBase = parseFloat(prompt("Enter the base of the triangle: "));
  const triangleHeight = parseFloat(prompt("Enter the height of the triangle:
"));
  const triangleArea = calculateTriangleArea(triangleBase, triangleHeight);
  console.log("Area of the triangle: "+triangleArea);

  const rectangleLength = parseFloat(prompt("Enter the length of the
rectangle: "));
  const rectangleWidth = parseFloat(prompt("Enter the width of the rectangle:
"));
  const rectangleArea = calculateRectangleArea(rectangleLength,
rectangleWidth);
  console.log("Area of the Rectangle: "+rectangleArea);

  const circleRadius = parseFloat(prompt("Enter the radius of the circle: "));
  const circleArea = calculateCircleArea(circleRadius);
  console.log("Area of the Circle: "+circleArea);
```

```
Enter the base of the triangle: 10
Enter the height of the triangle: 20
Area of the triangle: 100
Enter the length of the rectangle: 100
Enter the width of the rectangle: 50
Area of the Rectangle: 5000
Enter the radius of the circle: 50
Area of the Circle: 7853.981633974483
```

Experiment No 2

```
const prompt=require('prompt-sync')();

function multTable(num) {
    for (let i = 1; i <= 10; i++) {
      const res = num * i;
      console.log(`${num} x ${i} = ${res}`);
    }
  }

  let n = prompt("Enter a Number");
  multTable(n);
```

```
Enter a Number6
6 x 1 = 6
6 x 2 = 12
6 x 3 = 18
6 x 4 = 24
6 x 5 = 30
6 x 6 = 36
6 x 7 = 42
6 x 8 = 48
6 x 9 = 54
6 x 10 = 60
```

Experiment No 3

3.1] Reverse A String

```
const prompt = require('prompt-sync')();

function revStr(input) {
    return input.split('').reverse().join('');
}

let str = prompt("Enter A String To Reverse")
let reversedStr = revStr(strToReverse);

console.log("Original String:", str);
console.log("Reversed String:", reversedStr);
```

```
Enter A String To Reverse omkar
Original String:  omkar
Reversed String: rakmo
```

3.2]Replace A Character

```
function replaceChars(input, search, replace) {
    return input.split(search).join(replace);
}

const str = "Programming is fun!";
const searchChar = 'm';
const replaceChar = 'X';

const replacedStr = replaceChars(str, searchChar, replaceChar);

console.log("Original String:", str);
console.log("Replaced String:", replacedStr);
```

```
Original String: Programming is fun!
Replaced String: PrograXXing is fun!
```

3.3]Checking Palindrome String

```
const prompt = require('prompt-sync')();

function revStr(input) {
    return input.split('').reverse().join('');
}

function isPalin(input) {
    let original = input.toLowerCase();
    console.log("Original String is : "+original);

    let reversed = revStr(original).toLowerCase();
    console.log("Reversed String is : "+reversed);

    if (original===reversed) {
        console.log("String Is Palindrome");
    } else {
        console.log("String Is Not Palindrome");
    }

}

let str = prompt("Enter A String : ")

isPalin(str);
```

```
Enter A String : omkar
Original String is : omkar
Reversed String is : rakmo
String Is Not Palindrome

Enter A String : madam
Original String is : madam
Reversed String is : madam
String Is Palindrome
```

Experiment No 4

Method 1: Using toUpperCase()

```
const string1 = 'JavaScript Program';
const string2 = 'javascript program';

const result = string1.toUpperCase() === string2.toUpperCase();

if(result) {
    console.log('The strings are similar.');
} else {
    console.log('The strings are not similar.');
}
```

```
The strings are similar.
```

Method 2: JS String Comparison Using RegEx

```
const string1 = 'JavaScript Program';
const string2 = 'javascript program';

const pattern = new RegExp(string1, "gi");

const result = pattern.test(string2)

if(result) {
    console.log('The strings are similar.');
} else {
    console.log('The strings are not similar.');
}
```

```
The strings are similar.
```

Method 3: Using localeCompare()

```
const string1 = 'JavaScript Program';
const string2 = 'javascript program';

const result = string1.localeCompare(string2, undefined, { sensitivity: 'base'
});

if(result == 0) {
    console.log('The strings are similar.');
} else {
    console.log('The strings are not similar.');
}
```

```
The strings are similar.
```