1. Multiplication Table:
- Write a function that prints a 10x10 multiplication table. The table should display the product of numbers from 1 to 10 in a grid format.

```
for(var i=1;i<=10;i++)
{
    for(var j=1;j<=10;j++)
    {
        console.log(i+"*"+j+"="+(i*j));
    }
    console.log("\n");
}
```

o/p:

```
1*1=1
1*2=2
1*3=3
1*4=4
1*5=5
1*6=6
1*7=7
1*8=8
1*9=9
1*10=10
2*1=2
2*2=4
2*3=6
2*4=8
2*5=10
2*6=12
2*7=14
2*8=16
2*9=18
2*10=20
3*1=3
3*2=6
3*3=9
3*4=12
3*5=15
3*6=18
3*7=21
3*8=24
3*9=27
3*10=30
4*1=4
4*2=8
4*3=12
4*4=16
4*5=20
4*6=24
4*7=28
4*8=32
4*9=36
4*10=40
```

```
5*1=5
5*2=10
5*3=15
5*4=20
5*5=25
5*6=30
5*7=35
5*8=40
5*9=45
5*10=50
6*1=6
6*2=12
6*3=18
6*4=24
6*5=30
6*6=36
6*7=42
6*8=48
6*9=54
6*10=60
7*1=7
7*2=14
7*3=21
7*4=28
7*5=35
7*6=42
7*7=49
7*8=56
7*9=63
7*10=70
8*1=8
8*2=16
8*3=24
8*4=32
8*5=40
8*6=48
8*7=56
8*8=64
8*9=72
8*10=80
9*1=9
9*2=18
9*3=27
9*4=36
9*5=45
9*6=54
9*7=63
9*8=72
9*9=81
9*10=90
10*1=10
10*2=20
10*3=30
10*4=40
```

```
10*5=50
10*6=60
10*7=70
10*8=80
10*9=90
10*10=100
```

2. Prime Numbers in a Range:

- Write a function that finds and prints all prime numbers between 1 and 100. A prime number is only divisible by 1 and itself.

```
var arr=[]
for(i=1;i<=100;i++)
{
    count=0;
    for (j=i;j>0;j--)
    {
        if(i%j == 0)
        {

        count=count+1;
        }
    }
        if (count==2)
            {
            arr.push(i);
            }
        }
console.log(arr);
```

3. FizzBuzz:

- Write a function that prints the numbers from 1 to 100. For multiples of three, print "Fizz" instead of the number. For multiples of five, print "Buzz". For numbers which are multiples of both three and five, print "FizzBuzz".

```
a=[]
for(let i=1;i<=100;i++)
{
    if((i%3 == 0) && (i%5 ==0))
    {
        a.push("FizzBuzz");
    }

    else if(i%3 == 0)
    {
        a.push("Fizz");
    }
    else if(i%5 == 0)
    {
        a.push("Buzz");
    }
     else
    {
        a.push(i);
    }

}
console.log(a);
```

o/p:

```
[
  1,      2,      'Fizz',     4,      'Buzz', 'Fizz',
  7,      8,      'Fizz',     'Buzz', 11,     'Fizz',
  13,     14,     'FizzBuzz', 16,     17,     'Fizz',
  19,     'Buzz', 'Fizz',     22,     23,     'Fizz',
  'Buzz', 26,     'Fizz',     28,     29,     'FizzBuzz',
  31,     32,     'Fizz',     34,     'Buzz', 'Fizz',
  37,     38,     'Fizz',     'Buzz', 41,     'Fizz',
  43,     44,     'FizzBuzz', 46,     47,     'Fizz',
  49,     'Buzz', 'Fizz',     52,     53,     'Fizz',
  'Buzz', 56,     'Fizz',     58,     59,     'FizzBuzz',
  61,     62,     'Fizz',     64,     'Buzz', 'Fizz',
  67,     68,     'Fizz',     'Buzz', 71,     'Fizz',
  73,     74,     'FizzBuzz', 76,     77,     'Fizz',
  79,     'Buzz', 'Fizz',     82,     83,     'Fizz',
  'Buzz', 86,     'Fizz',     88,     89,     'FizzBuzz',
  91,     92,     'Fizz',     94,     'Buzz', 'Fizz',
  97,     98,     'Fizz',     'Buzz'
]
```

4.

5. Pattern Printing:
- Write a function that prints the following pattern for a given number of rows:

```
*


*


*
```

ans:

```
n=5
var a=function(n)
{
for(var i=0;i<n;i++)
{
  console.log("*");
}
}
console.log(a(n));
```

```
o/p:
*
*
*
*
*

6.
7.
8.
9.
10. (still trying)
```