**1**. function Print\_Number(){

for(i=1;i<=100;i++){

if(i%3==0)

{

console.log("Fizz")

}

else if(i%5==0)

{

console.log("Buzz")

}

else if(i%3==0 & i%5==0)

{

console.log("FizzBuzz")

}

else{

console.log(i)

}

}

}

Print\_Number()

**2**. function arithmatic\_operation(){

let st=prompt()

a1=parseInt(prompt("enter first number"))

a2=parseInt(prompt("enter second number"))

if(st.toUpperCase() == "ADDITION"){

return a1+a2

}

else if(st.toUpperCase() == "SUBTRACTION"){

return a1-a2

}

}

arithmatic\_operation()

**3**. function flattened\_array(arr){

let result = []

for(let item of arr){

if (Array.isArray(item)){

result.push(flattened\_array(item))

}

else{

result.push(item)

}

}

return result

}

Const arr=[1, [2, [3, 4]], 5, [6]]

flattened\_array(arr)

4. function anagrams(str1,str2){

let s=0

if(str1.length == str2.length){

for( i of str1 ){

if (i in str2){

s = s+1

}

}

if (s == str1.length){

console.log("anagrams string")

}

}

else{

console.log("Not anagrams string")

}

}

let str1=prompt()

let str2=prompt()

anagrams(str1,str2)

5. function remove\_duplicate(arr){

let arr1=[]

for(i in arr){

if(!(i in arr1)){

arr1.push(i)

}

}

return arr1

}

let arr=[1,2,3,4,1,4]

remove\_duplicate(arr)

6. function capitalize(str1){

let final=""

final += str1[0].toUpperCase()

for(i=1;i<str1.length;i++){

console.log(str1[i])

if(str1[i-1] === " ")

{

final += str1[i].toUpperCase()

}

else{

final += str1[i]

}

}

return final

}

let str1="how can i make capital"

console.log(capitalize(str1))

7. function fibonacci(n){

let a=0

let b=1

let next

let f\_series=[]

for (i=0;i<n+1;i++){

if(i<=0){

console.log("Enter number grater than zero")

}

else if(i==1){

f\_series.push(a)

}

else if(i==2){

f\_series.push(b)

}

else{

next=a+b

a=b

b=next

f\_series.push(next)

}

}

return f\_series

}

let s1=4

console.log(fibonacci(s1))

8. class HashMap{

constructor (intialCapacity = 8){

this.bucketArray = new Array(intialCapacity);

this.bucketSize = 0;

}

\_hash(key){

let hash = 0;

for(let i = 0; i < key.length(i); i++){

hash += key.charCodeAt(i);

}

return hash % this.bucketArray.length;

}

get(key){

const index = this.\_hash(key);

if (!this.bucketArray[index]) return undefined;

for(const pair of this.bucketArray[index]){

if(pair.key === key ){

return pair.value

}

}

return undefined

}

put(key){

const index = this.\_hash(key);

if(! this.bucketArray[index]){

this.bucketArray[index] = [];

}

this.bucketArray[index].push({key, value})

this.bucketSize++;

}

remove(key){

const index = this.\_hash(key);

if(!this.bucketArray[index]) return undefined

const bucket = this.bucketArray[index];

let removedIndex = -1;

for(let i = 0; i < bucket.length; i++){

if(this.bucket[i].key === key){

removedIndex = i;

break;

}

}

if(removedIndex !== -1){

bucket.splice(removedIndex,1;)

this.bucketSize--;

}

return undefined

}

}

const myHashMap = new HashMap();

myHashMap.put("name","Ramesh");

myHashMap.put("age",56);

console.log(myHashMap.get("name"))

console.log(myHashMap.get("age"))

console.log(myHashMap.remove("age"))

console.log(myHashMap.get("age"))

9. function filter\_even(arr){

let filtered\_array=[]

for(let i of arr){

if(i%2 !== 0){

filtered\_array.push(i)

}

}

return filtered\_array

}

let arr=[1,2,3,4,5]

console.log(filter\_even(arr))

10. function capitalize(str1){

str1=str1.toLowerCase()

let final=""

final += str1[0].toUpperCase()

for(i=1;i<str1.length;i++){

console.log(str1[i])

if(str1[i-1] === " " | str1[i-1] === ".")

{

final += str1[i].toUpperCase()

}

else{

final += str1[i]

}

}

return final

}

let str1="hoW can i mAke tiTLECase."

console.log(capitalize(str1))