**Assignment-2**

1. Symbols

Code-

<!DOCTYPE html>

<html>

    <body>

        <script>

            var fib\_generator=function \*(){

                var current = 0, next = 1;

                while(true)

                {

                    [next, current] = [next+current, next];

                    yield current;

                }

            }

            var fib = fib\_generator();

            for(var i = 0; i < 10; i++){

                console.log(fib.next().value);

            }

        </script>

    </body>

</html>

1. Iterators

Code-

<!DOCTYPE html>

<html>

    <head>

        <meta charset="utf-8">

    </head>

    <body>

        <script>

            let iterable=[];

            function getArmstrong(number) {

                var n=3;

                for (var i=0;i<10;i++) {

                    for(var j=0;j<10;j++) {

                        for(var k=0;k<10;k++) {

                            var pow=(Math.pow(i,n)+Math.pow(j,n)+Math.pow(k,n));

                            var plus=(i\*100+j\*10+k);

                            if(pow==plus) {

                                console.log(pow);

                            }

                        }

                    }

                }

            }

            getArmstrong();

        </script>

    </body>

</html>

Output-

0

1

153

370

371

407

3. Generators

Code-

<!DOCTYPE html>

<html>

    <head>

        <meta charset="utf-8">

    </head>

    <body>

        <script>

            let iterable=[];

            function\*getArmstrong(number) {

                yield 1;

                var n=3;

                for (var i=0;i<10;i++) {

                    for(var j=0;j<10;j++) {

                        for(var k=0;k<10;k++) {

                            var pow=(Math.pow(i,n)+Math.pow(j,n)+Math.pow(k,n));

                            var plus=(i\*100+j\*10+k);

                            if(pow==plus) {

                                console.log(pow);

                            }

                        }

                    }

                }

            }

            let myGen=getArmstrong();

            console.log(myGen.next());

            console.log(myGen.next());

        </script>

    </body>

</html>

Output-

{value: 1, done: false}

done: false

value: 1

[[prototype]]: object

0

1

153

370

371

407

{value: undefined, done: true}

4. Collections

Code-

<!DOCTYPE html>

<html>

    <body>

        <script>

            let chatroom1 = new Map();

            let chatroom2 = new Map();

            let user1 = new Set(["hii"]);

            let user2 = new Set(["hello"]);

            let user3 = new Set(["hello world"]);

            let user4 = new Set(["c2u1","whtsapp"]);

            let user5 = new Set(["c2u2","what are you pursing"]);

            let user6 = new Set(["c2u3","how is your day?"]);

            chatroom1.set("user1 tinku",user1);

            chatroom1.set("user2 ammu",user2);

            chatroom1.set("user3 senorita",user3);

            chatroom2.set("user1 sonu",user4);

            chatroom2.set("user2 Faraha",user5);

            chatroom2.set("user3 rishi",user6);

            function printusersofchat(chatroom){

                for(let key of chatroom.keys()){

                    console.log("\n",key);

                }

            }

            function printchatsofchat(chatroom){

                let chats = new Set();

                for(let key of chatroom.values()){

                    key.forEach(chats.add,chats);

                }

                console.log("\n",chats);

            }

            console.log("\nUsers of chatroom 1 :");

            printusersofchat(chatroom1);

            console.log("\nUsers of chatroom 2 :");

            printusersofchat(chatroom2);

            console.log("\n Chats from chatroom 1 :");

            printchatsofchat(chatroom1);

            console.log("\n Chats from chatroom 2 :");

            printchatsofchat(chatroom2);

        </script>

    </body>

</html>