**Write a JavaScript to accept a number from user & check whether it is Armstrong number or not. (Hint: Armstrong number means the summation of cubes of all the three digits of the number should be exactly to the number) E.g., 153= (1\*1\*1) +(5\*5\*5) +(3\*3\*3)**

function isArmstrongNumber(number) {

let sum = 0;

let temp = number;

const digits = number.toString().length;

while (temp > 0) {

let digit = temp % 10;

sum += Math.pow(digit, digits);

temp = Math.floor(temp / 10);

}

return sum === number;

}

// Prompting the user to enter a number

const userInput = parseInt(prompt("Enter a number:"));

// Checking if the entered number is an Armstrong number or not

if (isArmstrongNumber(userInput)) {

console.log(`${userInput} is an Armstrong number.`);

} else {

console.log(`${userInput} is not an Armstrong number.`);

}

**Write JavaScript code to print following pattern.**

**1**

**0 1**

**1 0 1**

**0 1 0 1**

**1 0 1 0 1**

function printPattern(rows) {

for (let i = 0; i < rows; i++) {

let rowPattern = "";

for (let j = 0; j <= i; j++) {

if ((i + j) % 2 === 0) {

rowPattern += "1 ";

} else {

rowPattern += "0 ";

}

}

console.log(rowPattern);

}

}

// Change the number of rows as per your requirement

const numberOfRows = 5;

printPattern(numberOfRows);

**Write a JS to print all number from 1-100 by adhering following condition: print only those number whose reverse of square and square of reverse are same. For ex. Square(12)=144, Reverse(144)=441 and Reverse(12)=21,Square(21)=441. so, final results of both the cases are same.Print all numbers following these cases between 1-100.**

function reverseNumber(num) {

let reversed = 0;

while (num > 0) {

reversed = reversed \* 10 + (num % 10);

num = parseInt(num / 10);

}

return reversed;

}

// Function to check if the reverse of square and square of reverse are same

function isReverseSquareEqual(num) {

const square = num \* num;

const reverse = reverseNumber(square);

const rv=reverseNumber(num)

const reverseSquare = rv \* rv;

return reverse === reverseSquare;

}