🔁 FOR LOOP – Practice Questions

🧪 Q1. Printing Student Roll Numbers

📘 Problem:

In a school, roll numbers start from 1 to 20. Print all the roll numbers.

```
for (let i = 1; i <= 20; i++) {
  console.log(`Roll No: ${i}`);
}
```

🧪 Q2. Display Squares of Numbers

📘 Problem:

Print the square of numbers from 1 to 10. Format: `"Square of 2 is 4"`

```
for (let i = 1; i <= 10; i++) {
  console.log(`Square of ${i} is ${i * i}`);
}
```

🧪 Q3. Sum of Even Numbers till 50

📘 Problem:

Find and print the sum of all even numbers from 1 to 50.

```
let sum = 0;
```

```
for (let i = 2; i <= 50; i += 2) {
  sum += i;
}
console.log("Sum of even numbers:", sum);
```

🔁 WHILE LOOP – Practice Questions

🧪 Q4. Battery Charging Display

📘 Problem:

You are building a UI for a battery. Show a charging level every 10% until it reaches 100%.

```
let level = 10;
while (level <= 100) {
  console.log(`Battery at ${level}%`);
  level += 10;
}
```

🧪 Q5. ATM PIN Retry

📘 Problem:

A user has 3 attempts to enter the correct ATM PIN (say 1234). If they enter the wrong pin, ask again. Stop after 3 attempts or when correct.

```
let attempts = 0;
const correctPIN = "1234";
```

```
let enteredPIN;

while (attempts < 3) {
  enteredPIN = prompt("Enter your ATM PIN:");
  if (enteredPIN === correctPIN) {
    console.log("PIN Accepted!");
    break;
  } else {
    console.log("Wrong PIN");
    attempts++;
  }
}
if (attempts === 3) {
  console.log("Card Blocked.");
}
```

🧪 Q6. Countdown Timer

📘 Problem:

Display countdown from 10 to 1 using a while loop.

```
let count = 10;
while (count > 0) {
  console.log(count);
  count--;
}
console.log("Time's up!");
```

🔁 DO...WHILE LOOP – Practice Questions

🧪 Q7. At Least One Attempt (Login Prompt)

📘 Problem:

Ask the user for a username. Show "Welcome" if not empty, or ask again (but ask at least once).

```
let username;
do {
  username = prompt("Enter your username:");
} while (!username);
console.log(`Welcome, ${username}!`);
```

🧪 Q8. Menu System Simulation

📘 Problem:

Simulate a menu with options (1. View, 2. Add, 3. Exit). Keep showing until user presses 3.

```
let choice;
do {
  choice = parseInt(prompt("1. View\\n2. Add\\n3. Exit\\nEnter your choice:"));
  switch (choice) {
    case 1:
      console.log("Viewing Items...");
      break;
    case 2:
      console.log("Adding Item...");
```

```
      break;
    case 3:
      console.log("Exiting...");
      break;
    default:
      console.log("Invalid choice!");
  }
} while (choice !== 3);
```

🧪 Q9. Dice Game Until 6

📘 Problem:

Keep rolling a dice (1–6) until the number 6 appears. Print each roll.

```
let roll;
do {
  roll = Math.floor(Math.random() * 6) + 1;
  console.log("You rolled:", roll);
} while (roll !== 6);
```

🧪 Q1. Print Multiplication Table of a Number

📘 Problem:

Take a number (e.g., 7) and print its multiplication table up to 10.

```
const num = 7;
for (let i = 1; i <= 10; i++) {
```

```
  console.log(`${num} x ${i} = ${num * i}`);
}
```

🧪 Q2. Keep Asking Age Until User Enters 18 or Above

📘 Problem:

Keep asking for the user's age until they enter a value >= 18. Show a welcome message.

```
let age;
do {
  age = parseInt(prompt("Enter your age:"));
} while (age < 18);
console.log("You are eligible!");
```

🧪 Q3. Calculate the Sum of Digits of a Number

📘 Problem:

Given a number (e.g., 1234), calculate the sum of its digits.

```
let num = 1234;
let sum = 0;

while (num > 0) {
  sum += num % 10;
  num = Math.floor(num / 10);
}
```

```
console.log("Sum of digits:", sum);
```

🧪 Q4. Print All Characters of a String One by One

📘 Problem:

Print each character of the string `"HELLO"` on a new line.

```
const str = "HELLO";
for (let i = 0; i < str.length; i++) {
  console.log(str[i]);
}
```

🧪 Q5. Find Factorial of a Number (e.g., 5!)

📘 Problem:

Calculate and print the factorial of `5`.

```
let num = 5;
let fact = 1;
for (let i = 1; i <= num; i++) {
  fact *= i;
}
console.log(`Factorial of ${num} is ${fact}`);
```

🧪 Q6. Simulate Password Prompt Until Correct

📘 Problem:

Keep asking the user for a password until they enter `"admin123"`.

```
let password;
do {
  password = prompt("Enter password:");
} while (password !== "admin123");

console.log("Access Granted.");
```

🧪 Q7. Count How Many Numbers Between 1–50 Are Divisible by 7

📘 Problem:

Print and count how many numbers between 1 and 50 are divisible by 7.

```
let count = 0;
for (let i = 1; i <= 50; i++) {
  if (i % 7 === 0) {
    console.log(i);
    count++;
  }
}
console.log("Total divisible by 7:", count);
```

🧪 Q8. Keep Rolling a Dice Until You Get an Odd Number

📘 Problem:

Keep simulating a dice roll until you get an odd number.

```
let roll;
do {
  roll = Math.floor(Math.random() * 6) + 1;
  console.log("Rolled:", roll);
} while (roll % 2 === 0);
```

🧪 Q9. Show All Years from 2000 to 2025

📘 Problem:

Print each year from 2000 to 2025.

```
for (let year = 2000; year <= 2025; year++) {
  console.log(year);
}
```

🧪 Q10. Countdown with Message: "Launching in X..."

📘 Problem:

Print a countdown from 5 to 1 with a launch message at the end.

```
let count = 5;
while (count > 0) {
  console.log(`Launching in ${count}...`);
  count--;
```

```
}
console.log("🚀 Launched!");
```

🧪 Q1. Hotel Room Booking Availability Tracker

📘 Problem:

A hotel has 5 floors, and each floor has 10 rooms. Print all available room numbers in the format: `"Floor 2 - Room 7"` but **skip Room 3 and Room 8 on each floor** as they are under maintenance.

```
for (let floor = 1; floor <= 5; floor++) {
  for (let room = 1; room <= 10; room++) {
    if (room === 3 || room === 8) continue;
    console.log(`Floor ${floor} - Room ${room}`);
  }
}
```

🧪 Q2. Reverse a Given Number and Check for Palindrome

📘 Problem:

Ask the user to enter a number (e.g., 121). Reverse it using a loop and check if the number is a palindrome.

```
let original = 121;
let temp = original;
let reverse = 0;

while (temp > 0) {
```

```
  reverse = reverse * 10 + (temp % 10);
  temp = Math.floor(temp / 10);
}

if (reverse === original) {
  console.log(`${original} is a palindrome`);
} else {
  console.log(`${original} is not a palindrome`);
}
```

🧪 Q3. Print Prime Numbers Between 2 and 50

📘 Problem:

You need to print all **prime numbers** from 2 to 50 using nested loops.

```
for (let i = 2; i <= 50; i++) {
  let isPrime = true;
  for (let j = 2; j <= Math.sqrt(i); j++) {
    if (i % j === 0) {
      isPrime = false;
      break;
    }
  }
  if (isPrime) {
    console.log(i);
  }
}
```

🧪 Q4. Find Armstrong Numbers Between 100 and 999

📘 Problem:

An Armstrong number is one where the **sum of the cubes of its digits equals the number** (e.g., 153 = $1^3 + 5^3 + 3^3$). Print all such 3-digit numbers.

```
for (let num = 100; num <= 999; num++) {
  let sum = 0;
  let temp = num;

  while (temp > 0) {
    let digit = temp % 10;
    sum += digit ** 3;
    temp = Math.floor(temp / 10);
  }

  if (sum === num) {
    console.log(num);
  }
}
```

🧪 Q5. Simulate Online Quiz Timer

📘 Problem:

Simulate a 10-second countdown for a quiz question. Every second, show a message like: `"10 seconds remaining..."`.

```
let time = 10;
```

```
while (time > 0) {
  console.log(`${time} seconds remaining...`);
  time--;
}
console.log("Time's up! Move to next question.");
```

🧪 Q6. Generate a Number Triangle Pattern

📘 Problem:

Generate the following triangle pattern using nested loops:

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

```
for (let i = 1; i <= 5; i++) {
  let row = "";
  for (let j = 1; j <= i; j++) {
    row += j + " ";
  }
  console.log(row.trim());
}
```

🧪 Q7. Simulate ATM Transaction Limit

📘 Problem:

A user can withdraw money up to 3 times in a day. If they try more than 3 times, stop them. After each transaction, ask if they want to continue.

```
let count = 0;
let choice;

do {
  count++;
  console.log(`Transaction ${count} done.`);
  if (count >= 3) {
    console.log("Daily limit reached.");
    break;
  }
  choice = prompt("Do you want to do another transaction? (yes/no)");
} while (choice === "yes");
```

🧪 Q8. Find the GCD of Two Numbers

📘 Problem:

Take two numbers (e.g., 18 and 24) and find their Greatest Common Divisor (GCD) using a loop.

```
let a = 18;
let b = 24;
let gcd = 1;

for (let i = 1; i <= Math.min(a, b); i++) {
  if (a % i === 0 && b % i === 0) {
```

```
    gcd = i;
  }
}
console.log(`GCD of ${a} and ${b} is ${gcd}`);
```

🧪 Q9. Password Retry with Lockout Simulation

📘 Problem:

Allow a user to enter a password (correct password is `"admin"`). Allow up to 5 attempts. If the user fails 5 times, show "Account Locked".

```
const correctPassword = "admin";
let attempt = 0;
let input;

while (attempt < 5) {
  input = prompt("Enter your password:");
  if (input === correctPassword) {
    console.log("Login Successful");
    break;
  } else {
    console.log("Wrong password");
    attempt++;
  }
}
if (attempt === 5) {
  console.log("Account Locked");
}
```

🧪 Q10. Check for Perfect Number

📘 Problem:

A number is **perfect** if the sum of its proper divisors equals the number (e.g., 6 → 1 + 2 + 3 = 6). Check whether a number is perfect or not.

```
let num = 28;
let sum = 0;

for (let i = 1; i < num; i++) {
  if (num % i === 0) {
    sum += i;
  }
}

if (sum === num) {
  console.log(`${num} is a perfect number`);
} else {
  console.log(`${num} is not a perfect number`);
}
```