

Eenvoudige Oefeningen:

1. **Tel van 1 tot 10:** Gebruik een `for`-loop om de getallen van 1 tot 10 op het scherm te schrijven.
2. **Tel van 10 tot 1:** Gebruik een `for`-loop om de getallen van 10 tot 1 op het scherm te schrijven.
3. **Print even getallen:** Gebruik een `for`-loop om de even getallen van 0 tot 20 te printen.
4. **Print oneven getallen:** Gebruik een `for`-loop om de oneven getallen van 1 tot 19 te printen.
5. **Print de tafel van 5:** Gebruik een `for`-loop om de tafel van 5 (5, 10, 15, ..., 50) te printen.
6. **Print getallen tussen 50 en 100:** Gebruik een `for`-loop om de getallen van 50 tot 100 te printen.
7. **Som van de eerste 10 getallen:** Gebruik een `for`-loop om de som van de eerste 10 getallen (1 tot 10) te berekenen.
8. **Product van de eerste 5 getallen:** Gebruik een `for`-loop om het product van de eerste 5 getallen (1, 2, 3, 4, 5) te berekenen.
9. **Print getallen die deelbaar zijn door 3:** Gebruik een `for`-loop om getallen van 1 tot 30 te printen die deelbaar zijn door 3.
10. **Print de tafel van een gegeven getal:** Vraag een getal aan de gebruiker en gebruik een `for`-loop om de tafel van dat getal te printen.

Middelmatige Oefeningen:

1. **Print machten van 2:** Gebruik een `for`-loop om de machten van 2 te printen, beginnend bij 1, totdat de macht groter is dan 1000.
2. **Tel af van een gegeven getal:** Vraag een getal aan de gebruiker en tel af naar 0 met een `for`-loop.
3. **Print een reeks van 1 tot n:** Vraag een getal `n` aan de gebruiker en print alle getallen van 1 tot `n`.
4. **Som van de eerste n getallen:** Vraag een getal `n` aan de gebruiker en bereken de som van de getallen van 1 tot `n`.
5. **Print de veelvouden van een gegeven getal:** Vraag een getal aan de gebruiker en print de veelvouden van dat getal tot 100.
6. **Som van de even getallen:** Gebruik een `for`-loop om de som van de even getallen van 1 tot 100 te berekenen.
7. **Som van de oneven getallen:** Gebruik een `for`-loop om de som van de oneven getallen van 1 tot 100 te berekenen.
8. **Print de kwadraten van getallen:** Gebruik een `for`-loop om de kwadraten van de getallen van 1 tot 10 te printen (bijv. 1, 4, 9, 16, ..., 100).
9. **Print getallen tussen 1 en n die deelbaar zijn door 5:** Vraag een getal `n` aan de gebruiker en print alle getallen van 1 tot `n` die deelbaar zijn door 5.
10. **Print het omgekeerde van een getal:** Vraag een getal aan de gebruiker en gebruik een `for`-loop om het omgekeerde van dat getal te printen (bijv. 1234 wordt 4321).

Geavanceerde Oefeningen:

1. **Print priemgetallen:** Gebruik een `for`-loop om alle priemgetallen tussen 1 en 100 te printen.
2. **Bereken de faculteit van een getal:** Vraag een getal aan de gebruiker en gebruik een `for`-loop om de faculteit (factorial) van dat getal te berekenen.
3. **Print getallen die een bepaald patroon volgen:** Print de getallenreeks 1, 2, 4, 8, 16, 32, ..., totdat de waarde groter is dan een door de gebruiker opgegeven getal.
4. **Tel de cijfers in een getal:** Vraag een getal aan de gebruiker en gebruik een `for`-loop om het aantal cijfers in dat getal te tellen.
5. **Bereken de som van de cijfers in een getal:** Vraag een getal aan de gebruiker en bereken de som van de cijfers in dat getal (bijv. bij 1234 is de som $1+2+3+4=10$).
6. **Vind de grootste deler van een getal:** Vraag een getal aan de gebruiker en gebruik een `for`-loop om de grootste deler van dat getal (anders dan het getal zelf) te vinden.
7. **Print de Fibonacci-reeks:** Gebruik een `for`-loop om de eerste 10 getallen van de Fibonacci-reeks te printen (bijv. 1, 1, 2, 3, 5, 8, ...).
8. **Som van de Fibonacci-reeks:** Bereken de som van de eerste n getallen in de Fibonacci-reeks, waarbij n door de gebruiker wordt ingevoerd.
9. **Controleer of een getal een priemgetal is:** Vraag een getal aan de gebruiker en gebruik een `for`-loop om te controleren of het een priemgetal is.
10. **Print een driehoek van sterretjes:** Gebruik een `for`-loop om een driehoek van sterretjes te printen, waarbij het aantal rijen door de gebruiker wordt opgegeven. Bijvoorbeeld, voor 5 rijen:

Eenvoudige Oefeningen:

1. Tel van 1 tot 10:

```
for (let i = 1; i <= 10; i++) {  
  console.log(i);  
}
```

Uitleg: De for-loop start bij 1 en blijft lopen zolang *i* kleiner of gelijk is aan 10, waarbij *i* na elke iteratie met 1 wordt verhoogd.

2. Tel van 10 tot 1:

```
for (let i = 10; i >= 1; i--) {  
  console.log(i);  
}
```

Uitleg: Hier begint de loop bij 10 en vermindert *i* elke keer met 1 totdat het gelijk is aan 1.

3. Print even getallen:

```
for (let i = 0; i <= 20; i += 2) {  
  console.log(i);  
}
```

Uitleg: De loop begint bij 0 en verhoogt *i* telkens met 2. Dit zorgt ervoor dat alleen even getallen worden geprint.

4. Print oneven getallen:

```
for (let i = 1; i < 20; i += 2) {  
  console.log(i);  
}
```

Uitleg: Deze loop begint bij 1 en verhoogt *i* met 2, waardoor alleen de oneven getallen worden weergegeven.

5. Print de tafel van 5:

```
for (let i = 1; i <= 10; i++) {  
  console.log(5 * i);  
}
```

Uitleg: De loop vermenigvuldigt het huidige iteratienummer met 5 om de tafel van 5 te printen.

6. Print getallen tussen 50 en 100:

```
for (let i = 50; i <= 100; i++) {  
  console.log(i);  
}
```

Uitleg: De loop begint bij 50 en verhoogt *i* met 1, totdat het 100 bereikt.

7. Som van de eerste 10 getallen:

```

let sum = 0;
for (let i = 1; i <= 10; i++) {
    sum += i;
}
console.log(sum);

```

Uitleg: De loop berekent de som door elke iteratie het getal *i* toe te voegen aan de variabele *sum*.

8. Product van de eerste 5 getallen:

```

let product = 1;
for (let i = 1; i <= 5; i++) {
    product *= i;
}
console.log(product);

```

Uitleg: Het product wordt berekend door elke iteratie het getal *i* te vermenigvuldigen met de variabele *product*.

9. Print getallen die deelbaar zijn door 3:

```

for (let i = 1; i <= 30; i++) {
    if (i % 3 === 0) {
        console.log(i);
    }
}

```

Uitleg: Deze loop print alleen de getallen tussen 1 en 30 die deelbaar zijn door 3 door een modulus-berekening ($i \% 3 === 0$).

10. Print de tafel van een gegeven getal:

```

let number = parseInt(prompt("Voer een getal in:"));
for (let i = 1; i <= 10; i++) {
    console.log(number * i);
}

```

Uitleg: Eerst wordt een getal gevraagd aan de gebruiker via *prompt*. De loop vermenigvuldigt dit getal met elke waarde van *i* van 1 tot 10 om de tafel van het getal te tonen.

Middelmatige Oefeningen:

1. Print machten van 2:

```
for (let i = 1; i <= 1000; i *= 2) {  
  console.log(i);  
}
```

Uitleg: De loop begint bij 1 en vermenigvuldigt *i* telkens met 2. Dit zorgt ervoor dat alleen machten van 2 geprint worden totdat *i* groter is dan 1000.

2. Tel af van een gegeven getal:

```
let number = parseInt(prompt("Voer een getal in:"));  
for (let i = number; i >= 0; i--) {  
  console.log(i);  
}
```

Uitleg: Hier wordt een getal van de gebruiker gevraagd en de loop telt van dat getal af naar 0.

3. Print een reeks van 1 tot n:

```
let n = parseInt(prompt("Voer een getal n in:"));  
for (let i = 1; i <= n; i++) {  
  console.log(i);  
}
```

Uitleg: Deze loop begint bij 1 en print alle getallen tot *n*, dat door de gebruiker is ingevoerd.

4. Som van de eerste n getallen:

```
let n = parseInt(prompt("Voer een getal n in:"));  
let sum = 0;  
for (let i = 1; i <= n; i++) {  
  sum += i;  
}  
console.log(sum);
```

Uitleg: De som wordt berekend door elke iteratie het huidige getal bij de variabele *sum* op te tellen.

5. Print de veelvouden van een gegeven getal:

```
let number = parseInt(prompt("Voer een getal in:"));  
for (let i = 1; i <= 100; i++) {  
  if (i % number === 0) {  
    console.log(i);  
  }  
}
```

Uitleg: Deze loop print de veelvouden van het getal dat door de gebruiker is ingevoerd, door de modulus-berekening te gebruiken (*i % number === 0*).

6. **Som van de even getallen:**

```
let sum = 0;
for (let i = 2; i <= 100; i += 2) {
  sum += i;
}
console.log(sum);
```

Uitleg: De loop begint bij 2 en verhoogt telkens met 2 om alleen de even getallen bij *sum* op te tellen.

7. **Som van de oneven getallen:**

```
let sum = 0;
for (let i = 1; i <= 100; i += 2) {
  sum += i;
}
console.log(sum);
```

Uitleg: Deze loop telt alle oneven getallen tussen 1 en 100 bij elkaar op.

8. **Print de kwadraten van getallen:**

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

Uitleg: De loop print het kwadraat van elk getal tussen 1 en 10 door *i* met zichzelf te vermenigvuldigen.

9. **Print getallen tussen 1 en n die deelbaar zijn door 5:**

```
let n = parseInt(prompt("Voer een getal n in:"));
for (let i = 1; i <= n; i++) {
  if (i % 5 === 0) {
    console.log(i);
  }
}
```

Uitleg: Deze loop print alleen de getallen die deelbaar zijn door 5 door de modulus-berekening te gebruiken.

10. Print het omgekeerde van een getal:

```
let number = parseInt(prompt("Voer een getal in:"));
let reversed = 0;

for (let i = number; i > 0; i = (i - (i % 10)) / 10) {
    let laatsteCijfer = i % 10; // Haal het laatste cijfer
    reversed = reversed * 10 + laatsteCijfer; // Voeg het laatste cijfer
    toe aan reversed
}

console.log(reversed);
```

Uitleg:

- De **initiatiefase** in de for-loop begint met `let i = number`, wat ervoor zorgt dat de loop begint met het ingevoerde getal.
- De **voorwaarde** is `i > 0`, wat betekent dat de loop doorgaat totdat `i` gelijk is aan 0.
- In de **updatefase** wordt `i` verminderd door telkens het laatste cijfer te verwijderen: `(i - (i % 10)) / 10`. Dit haalt het laatste cijfer weg.
- Binnen de loop wordt het omgekeerde getal opgebouwd door `reversed` telkens te vermenigvuldigen met 10 en het laatste cijfer (`i % 10`) toe te voegen.

Op deze manier wordt het omgekeerde van het getal berekend met een volledige for-loop.

Geavanceerde Oefeningen:

Hier zijn de oplossingen in JavaScript voor de oefeningen, zonder gebruik te maken van functies of arrays:

1. Print priemgetallen tussen 1 en 100:

```
for (let i = 2; i <= 100; i++) {  
  let isPrime = true;  
  for (let j = 2; j < i; j++) {  
    if (i % j === 0) {  
      isPrime = false;  
      break;  
    }  
  }  
  if (isPrime) {  
    console.log(i);  
  }  
}
```

Uitleg: De buitenste loop gaat door elk getal van 2 tot 100. De binnenste loop controleert of het getal *i* deelbaar is door een getal kleiner dan zichzelf. Als dat zo is, is het geen priemgetal.

2. Bereken de faculteit van een getal:

```
let number = parseInt(prompt("Voer een getal in:"));  
let factorial = 1;  
for (let i = 1; i <= number; i++) {  
  factorial *= i;  
}  
console.log(factorial);
```

Uitleg: De loop vermenigvuldigt elke waarde van *i* met de variabele *factorial*, beginnend bij 1.

3. Print getallen die een patroon volgen (1, 2, 4, 8, ...):

```
let limit = parseInt(prompt("Voer een limiet in:"));  
for (let i = 1; i <= limit; i *= 2) {  
  console.log(i);  
}
```

Uitleg: De loop begint bij 1 en vermenigvuldigt elke keer *i* met 2 totdat het getal groter is dan de door de gebruiker opgegeven limiet.

4. Tel de cijfers in een getal:

```
let number = parseInt(prompt("Voer een getal in:"));  
let count = 0;  
for (let i = number; i > 0; i = (i - (i % 10)) / 10) {  
  count++;  
}  
console.log(count);
```

Uitleg: De loop verwijdert telkens het laatste cijfer van *i* en telt hoe vaak dit gebeurt totdat *i* gelijk is aan 0.

5. **Bereken de som van de cijfers in een getal:**

```
let number = parseInt(prompt("Voer een getal in:"));
let sum = 0;
for (let i = number; i > 0; i = (i - (i % 10)) / 10) {
    sum += i % 10;
}
console.log(sum);
```

Uitleg: De loop haalt telkens het laatste cijfer van *i* en voegt dit toe aan de variabele *sum*, totdat het getal volledig verwerkt is.

6. **Vind de grootste deler van een getal:**

```
let number = parseInt(prompt("Voer een getal in:"));
let grootsteDeler = 1;
for (let i = 1; i < number; i++) {
    if (number % i === 0) {
        grootsteDeler = i;
    }
}
console.log(grootsteDeler);
```

Uitleg: De loop controleert voor elk getal kleiner dan het ingevoerde getal of het een deler is. Als dat zo is, wordt de grootste deler bijgewerkt.

7. **Print de Fibonacci-reeks:**

```
let a = 1, b = 1;
console.log(a);
console.log(b);
for (let i = 3; i <= 10; i++) {
    let volgende = a + b;
    console.log(volgende);
    a = b;
    b = volgende;
}
```

Uitleg: De eerste twee Fibonacci-getallen zijn 1. Vervolgens berekent de loop de volgende getallen in de reeks door telkens de som van de twee voorgaande getallen te nemen.

8. **Som van de Fibonacci-reeks:**

```
let n = parseInt(prompt("Voer het aantal Fibonacci-getallen in:"));
let a = 1, b = 1, sum = a + b;
for (let i = 3; i <= n; i++) {
    let volgende = a + b;
    sum += volgende;
    a = b;
    b = volgende;
}
console.log(sum);
```

Uitleg: De som van de Fibonacci-reeks wordt berekend door telkens de nieuwe Fibonacci-waarde toe te voegen aan de variabele *sum*.

9. Controleer of een getal een priemgetal is:

```
let number = parseInt(prompt("Voer een getal in:"));
let isPrime = true;
for (let i = 2; i < number; i++) {
  if (number % i === 0) {
    isPrime = false;
    break;
  }
}
if (isPrime) {
  console.log(number + " is een priemgetal.");
} else {
  console.log(number + " is geen priemgetal.");
}
```

Uitleg: De loop controleert of het getal deelbaar is door een getal kleiner dan zichzelf. Als dat zo is, is het geen priemgetal.

10. Print een driehoek van sterretjes:

```
let rows = parseInt(prompt("Voer het aantal rijen in:"));
for (let i = 1; i <= rows; i++) {
  let line = '';
  for (let j = 1; j <= i; j++) {
    line += '*';
  }
  console.log(line);
}
```

Uitleg: De buitenste loop bepaalt het aantal rijen. De binnenste loop voegt voor elke rij het juiste aantal sterretjes toe aan de variabele *line*, die vervolgens geprint wordt.