Working with loops

LESSON OBJECTIVES

- Understand iteration
- Explore loop types
- Functional illustration

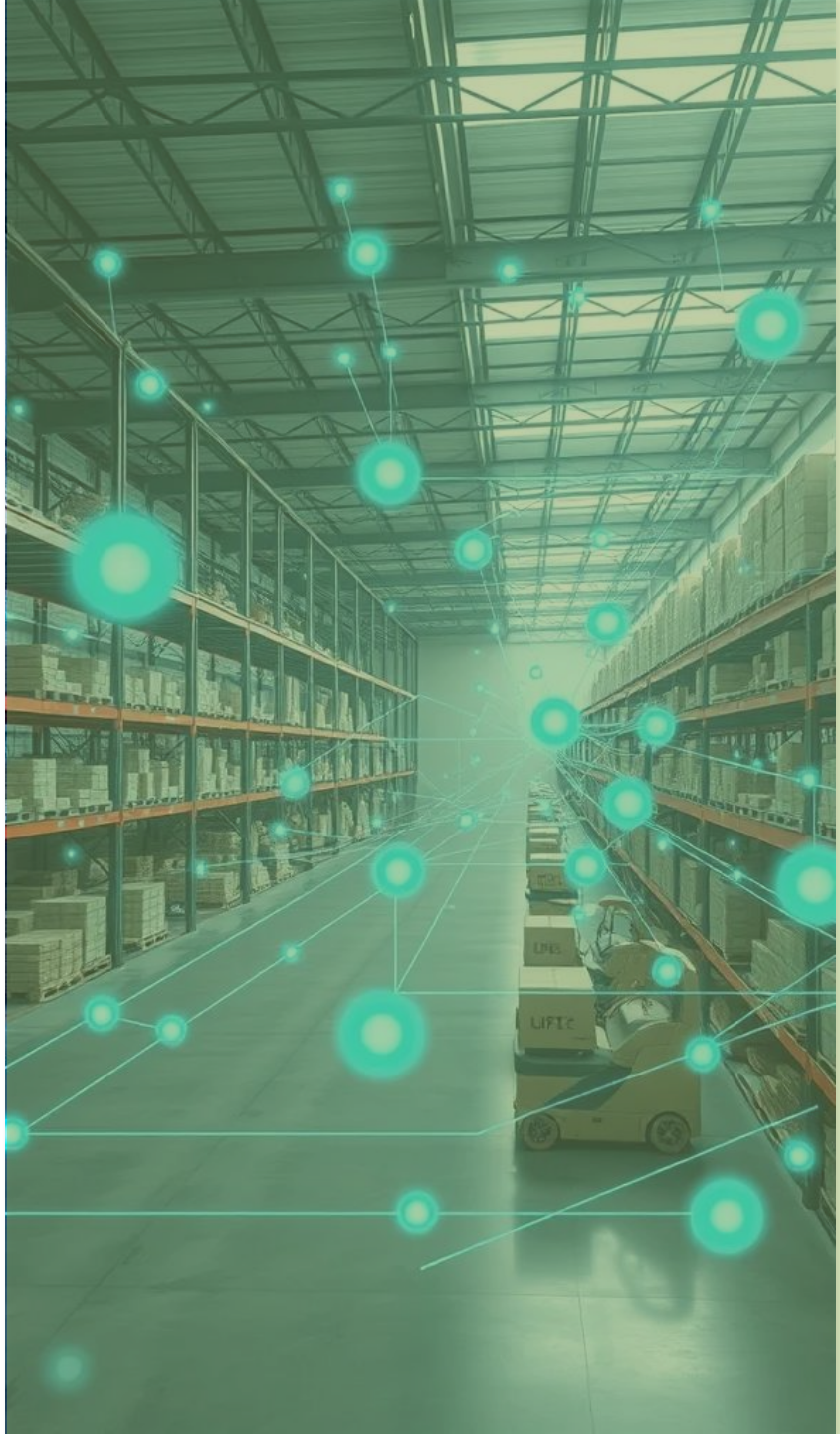
Iteration

Iteration in programming refers to the process of repeating a set of instructions multiple times. i.e. executing a block of code repeatedly until a certain condition is met.

Types of loops

for loop

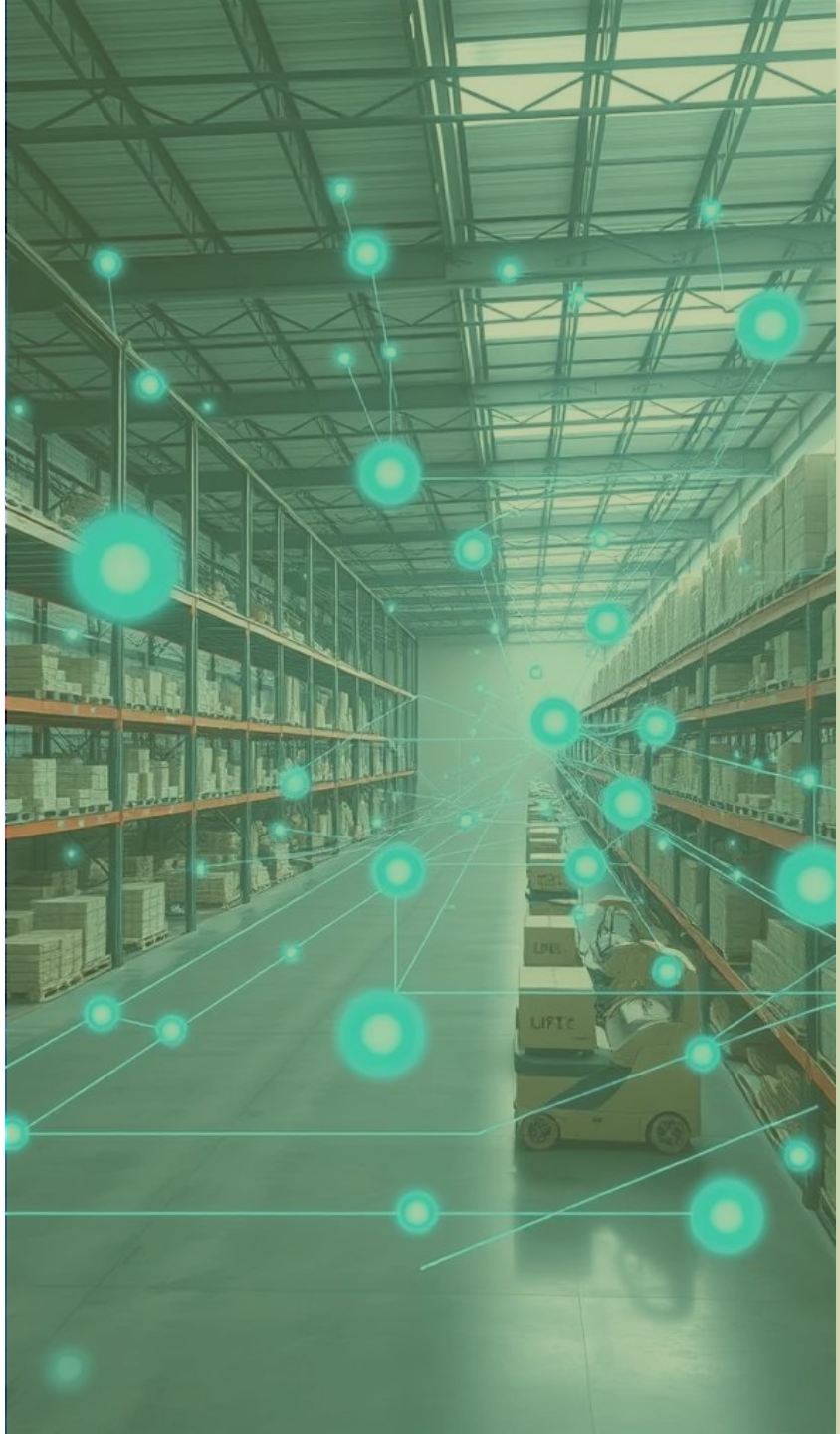
while loop



for loops

Typically used when you know the number of iterations in advance.

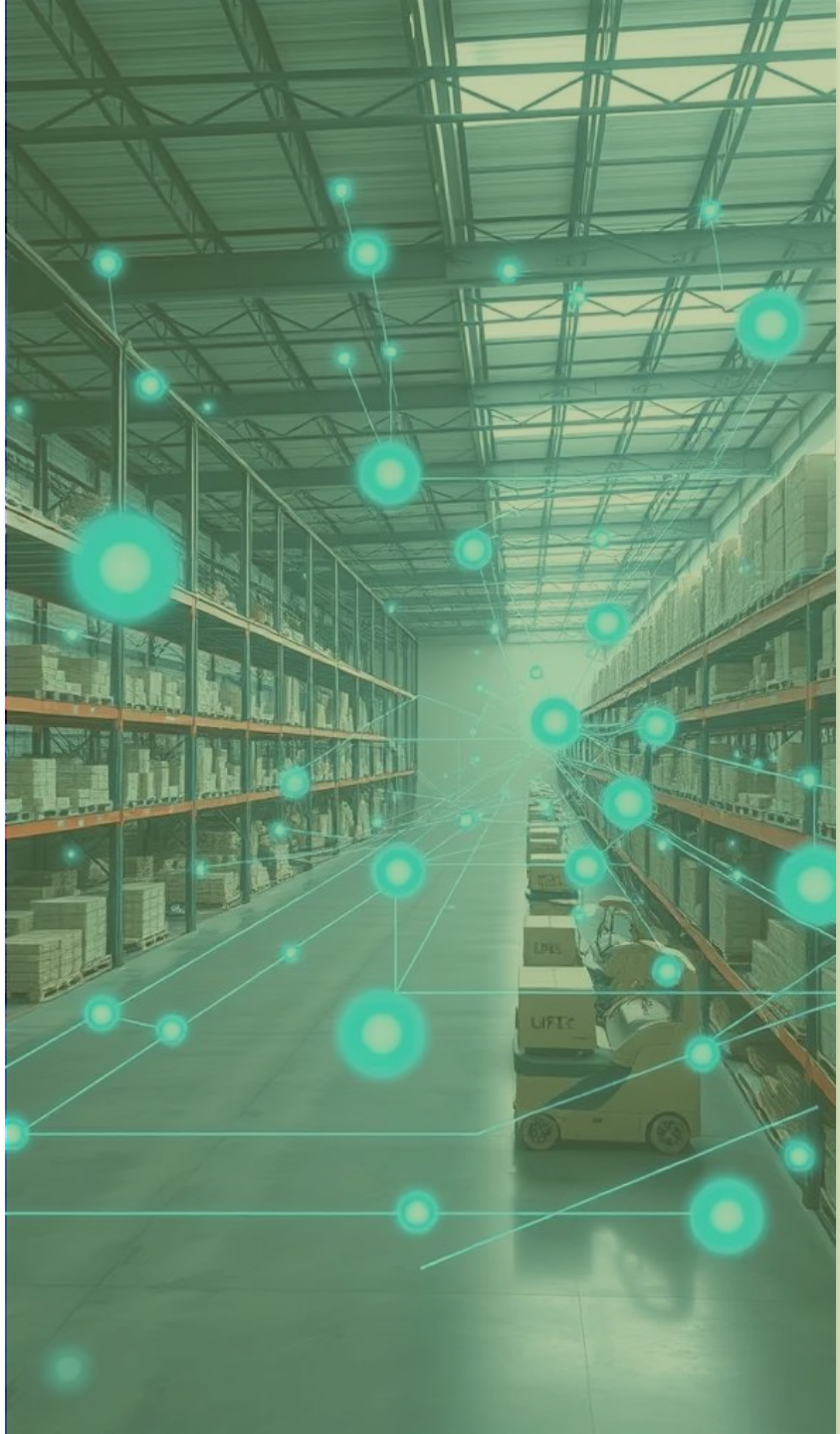
- **Structure**: Iterates over a sequence (list, tuple, string, etc.) or range
- **Initialization**: Loop variable is automatically initialized and updated
- **Use cases**: Best for iterating over collections or known ranges
- **Control flow:** Automatically terminates when it reaches the end of the sequence



while loops

Used when you want to repeat until a condition is met, number of iterations may be unknown.

- **Structure**: Continues as long as a boolean condition is true
- **Initialization**: You must initialize and update variables manually
- **Use cases**: Ideal for situations where you need to loop until a condition changes
- **Control flow:** Continues until the condition becomes false (risk of infinite loops if not managed properly)



for loop example

```
for number in range(1, 6):
    print(number)
```

while loop example

```
count = 0
while count < 5:
    print(count)
    count += 1
```

Functional illustration