



# Iterator Pattern

## LF10 Pattern Workshops

Mika  
Bomm

Jan-Henrik  
Ammer

Itch BS14

2025-09-08

# 1. What are patterns ?

---

# 1.1 What are Programming Patterns?

## 1. What are patterns ?



Problem	Pattern	Benefits
<b>Repeating coding challenges that many developers face</b> <ul style="list-style-type: none"><li>• Hard to maintain</li><li>• Reinventing the wheel</li></ul>	<b>Proven solution that works</b> <ul style="list-style-type: none"><li>• Reusable blueprint</li><li>• Best practices</li><li>• Tested by many developers</li></ul>	<ul style="list-style-type: none"><li>✓ Better code quality</li><li>✓ Easier maintenance</li><li>✓ Team communication</li><li>✓ Faster development</li><li>✓ Proven solutions</li></ul>

Patterns = Reusable solutions that make programming easier and more reliable

# 1.2 difrent typs of paterns ?

t

## 1. What are patterns ?



Let us (Lettuce)  begin

# 1.3 Prerequisites

From now on this is the Code we are sharing with every example we provide

1. What are patterns ?



```
1 fn main() {  
2     let numbers = vec![1, 2, 3, 4, 5, 6, 7, 8, 9, 10];  
3     let words = vec!["apple", "banana", "cherry", "date",  
4         "elderberry"];  
}
```

 Rust

# 1.4 What can we do with the Iterators

## 1. What are patterns ?

### 1.4.1 Basic Iterations



```
1 // easy
2 println!("{:?}", numbers); //outputs: 1 2 3 4 5 6 7 8 9 10
3 // complex
4 let output: String = numbers.iter().map(|x|
  x.to_string()).collect::
```

 Rust

### 1.4.2 Transformations

```
1 let doubled: Vec<i32> = numbers.iter().map(|x| x *
  2).collect();
2 println!("{:?}", doubled);
3 //outputs: [2, 4, 6, 8, 10, 12, 14, 16, 18, 20]
```

 Rust

# 1.4 What can we do with the Iterators

## 1. What are patterns ?

### 1.4.3 Filtering



```
1 let evens: Vec<i32> = numbers.iter().filter(|&x| x % 2 ==  
0).copied().collect();  
2 println!("{:?}", evens);  
3 //outputs: [2, 4, 6, 8, 10]
```

 Rust

### 1.4.4 Aggregating

```
1 let sum = numbers.iter().fold(0, |total, &x| total + x);  
2 println!("{}", sum);  
3 //outputs: 55
```

 Rust



left

middle

right

top

bottom

Lorem ipsum  
dolor sit amet,  
consectetur  
adipiscing elit.

Lorem ipsum dolor sit amet,  
consectetur adipiscing elit.

Lorem ipsum  
dolor sit amet,  
consectetur  
adipiscing elit.

Lorem ipsum  
dolor sit amet,  
consectetur  
adipiscing elit.

Lorem ipsum dolor sit amet,  
consectetur adipiscing elit.

Lorem ipsum  
dolor sit amet,  
consectetur  
adipiscing elit.

Lorem ipsum  
dolor sit amet,  
consectetur  
adipiscing elit.

Lorem ipsum dolor sit amet,  
consectetur adipiscing elit.

Lorem ipsum  
dolor sit amet,  
consectetur  
adipiscing elit.