

Iterator Pattern

LF10 Pattern Workshops

Mika Jan-Henrik Bomm Ammer

Itch BS14

2025-09-08

1. What are patterns?

1.1 What are Programming Patterns?

1. What are patterns?



Problem

challenges that m that works

developers face

- Hard to maintain
- Reinventing the

Pattern

Repeating coding Proven solution t

- Reusable bluepring
- Best practices
- Tested by many c

Benefits

- ✓ Better code quality
- ✓ Easier maintenance
- ✓ Team communication
- ✓ Faster development
- ✓ Proven solutions

Patters = Reusable solutions that make ro rammin easier and more reliable

1.2 Difrend typs of paterns?

t

1. What are patterns?



Let us (Lettuce) ይ begin

1.3 Prerequisites

1. What are patterns?

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



1.4 What can we do with the Iterators

1. What are patterns?

1.4.1 Basic Iterations



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]
```

1.4 What can we do with the Iterators

1. What are patterns?

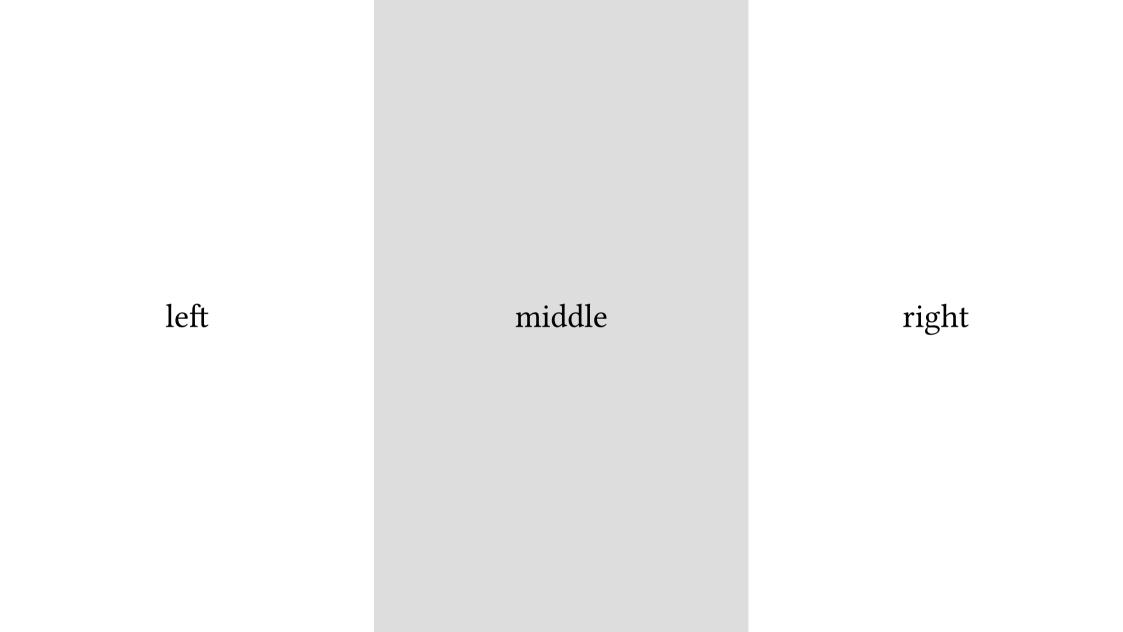
1.4.3 Filtering



```
let evens: Vec<i32> = numbers.iter().filter(|&x| x % 2 ==
0).copied().collect();
println!("{:?}", evens);
//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
```



top

bottom

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

consectetur
adipiscing elit.

Lorem ipsum
dolor sit amet,
consectetur
adipiscing elit.

Lorem ipsum