

If Let Expression

This lesson teaches the if let expression in Rust

We'll cover the following

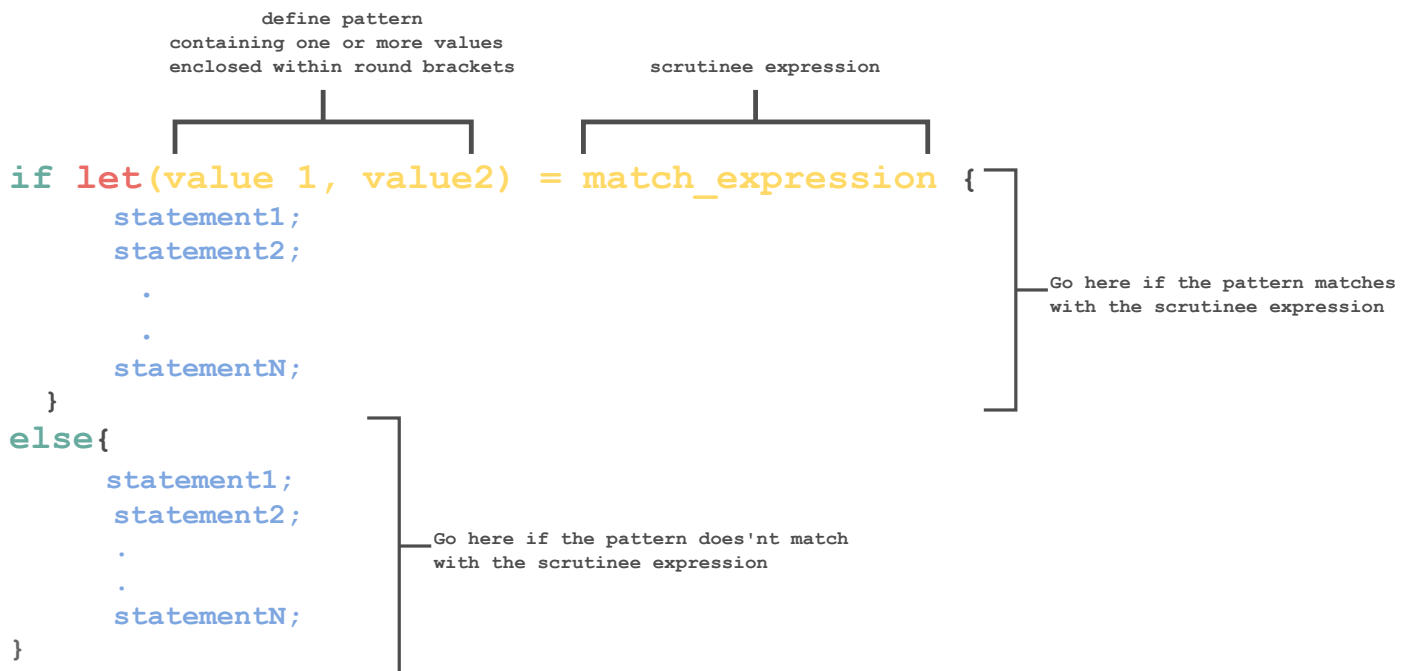
- What Is an if let Expression?
- Syntax
- Examples
 - Case 1: When the Pattern is Matched
 - Case 2: When the Pattern is Not Matched
 - Case 3: When the Pattern is Replaced With _
- Quiz

What Is an `if let` Expression?

`if let` is a conditional expression that allows pattern matching. The block of code in the construct executes if the pattern in the condition matches with that of `scrutinee` expression.

Syntax

The `if let` expression begins with an `if` followed by a `let` and then a pattern having values enclosed within round brackets. Then an equal to (`=`) followed by a `scrutinee` expression. Then there is a block of code enclosed within braces `{ }`. Then there is also an optional `else` block after this.



Note: When it says *matching of pattern*, it means that the defined pattern has the same number of values as that of the scrutinee expression.

Examples

The following examples show how the different cases of `if let` expression can work:

Case 1: When the Pattern is Matched

In the example below the defined pattern matches with the scrutinee expression.

```
fn main() {  
    // define a scrutinee expression  
    let course = ("Rust", "beginner", "course");  
    // pattern matches with the scrutinee expression  
    if let ("Rust", "beginner", "course") = course {  
        println!("Wrote all values in pattern to be matched with the scrutinee expression");  
    } else {  
        // do not execute this block  
        println!("Value unmatched");  
    }  
}
```



If the first value or second value matches, it can guess the third value.

```
fn main() {  
    // define a scrutinee expression  
    let course = ("Rust", "beginner", "course");  
    // pattern matches with the scrutinee expression  
    if let ("Rust", "beginner", c) = course {  
        println!("Wrote first two values in pattern to be matched with the scrutinee expression : ");  
    }  
    else {  
        // do not execute this block  
        println!("Value unmatched");  
    }  
}
```



If the first value matches, it can guess the other two values.

```
fn main() {  
    // define a scrutinee expression  
    let course = ("Rust", "beginner", "course");  
    // pattern matches with the scrutinee expression  
    if let ("Rust", c, d) = course {  
        println!("Wrote one value in pattern to be matched with the scrutinee expression. Guessed values are ");  
    } else {  
        // do not execute this block  
        println!("Value unmatched");  
    }  
}
```



Case 2: When the Pattern is Not Matched

In the example below, the defined pattern does not match with the scrutinee expression so the statement in the else block gets executed.

```
fn main() {  
    // define a scrutinee expression  
    let course = ("Rust", "beginner");  
    // pattern does not match with the scrutinee expression  
    if let ("Java", c) = course {  
        println!("Course is {}", c);  
    } else {  
        // execute this block  
        println!("Value unmatched");  
    }  
}
```



Case 3: When the Pattern is Replaced With `_`

In the example below, the pattern is not defined. Rather it is replaced with an `_`. In this case, the statement within the if let block executes.

Note: A **warning**, `△`, is generated by the compiler because the Rust compiler complains that it doesn't make sense to use **if let** with an **irrefutable pattern**.

```
fn main() {  
    // no pattern is define  
    if let _ = 10 {  
        println!("irrefutable if-let pattern is always executed");  
    }  
}
```



Quiz

Test your understanding of `if let` Expressions in Rust.

Quick Quiz on If Let Expression!

1

What is the output of the following code?

```
fn main() {  
    let course = ("Rust", "beginner", "course");  
    if let ("Rust", "beginer", "course") = course {  
        println!("Wrote all values in pattern to be matched with the scrutinee expression");  
    } else {  
        println!("Value unmatched");  
    }  
}
```

```
}
```

2



What is the output of the following code?

```
fn main() {  
    // no pattern is defined  
    if let _ = 10 {  
        println!("irrefutable if-let pattern");  
    }  
}
```

[Retake Quiz](#)

Now that you have learned about `if let`, learn about the `Match` statement in the next lesson.