

Nested Loops

This lesson discusses nested loops in Rust.

We'll cover the following ^

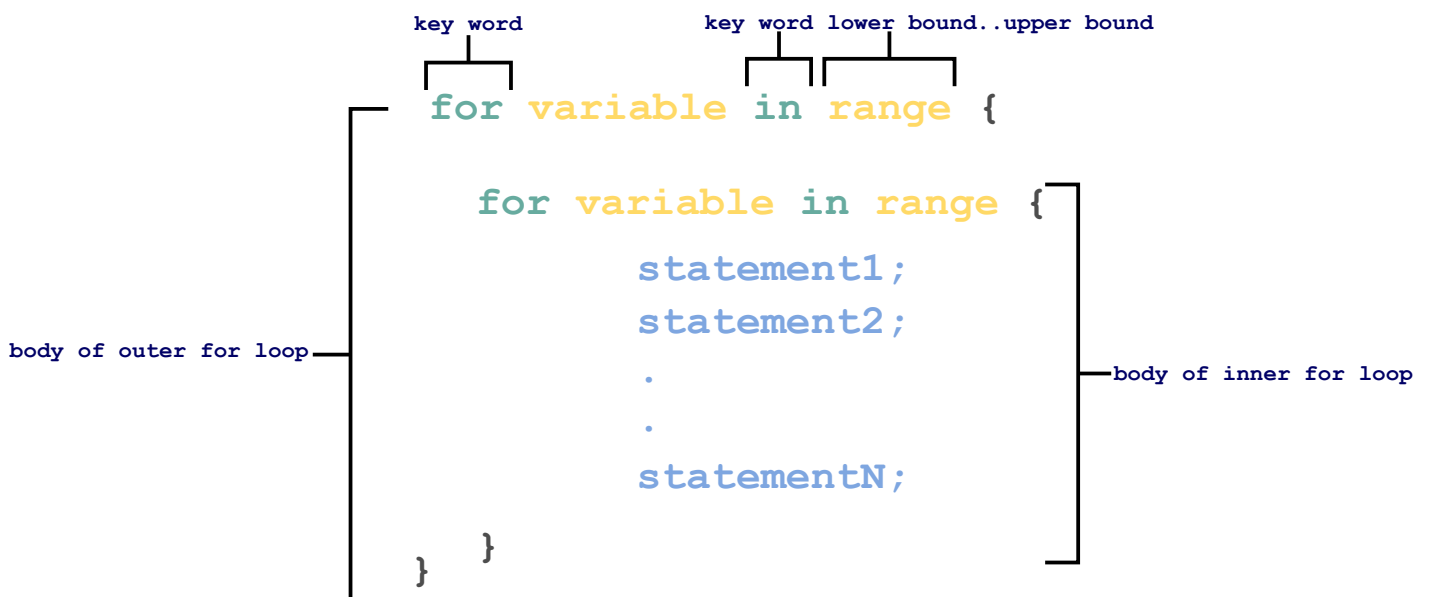
- What Is a Nested Loop?
- Syntax
- Example
 - Explanation

What Is a Nested Loop?

A nested loop is a loop within a loop.

Syntax

Here, a `for` loop *nested* inside a `for` loop. However, any loop can be nested inside any loop. The general syntax is:



Example

The following code prints a multiplication table of 1-5 using a nested `for` loop.

```
fn main() {
  for i in 1..5{ //outer loop
    println!("Multiplication Table of : {}", i);
    for j in 1..5 { // inner loop
      println!("{}", i * j);
    }
  }
}
```



Explanation

Outer **for** Loop

- A **for** loop is defined on **line 2**
- The loop takes **i** as an iterator that iterates over values from 1 to 5.

Inner **for** Loop

- A **for** loop is defined on **line 4**
- The loop takes **j** as an iterator that iterates over values from 1 to 5.
- For each iteration of the outer loop, the inner loop runs four times while printing the product of variables **i** and **j**.

i	j	Output
1	1	1*1=1
	2	1*2=2
	3	1*3=3
	4	1*4=4
2	1	2*1=2
	2	2*2=4
	3	2*3=6
	4	2*4=8
3	1	3*1=3
	2	3*2=6
	3	3*3=9
	4	3*4=12
4	1	4*1=4
	2	4*2=8
	3	4*3=12
	4	4*4=16

Sometimes, you might need to break the outer loop instead of the inner loop. So how can you specify which loop you are referring to? You can use a loop label.

Learn about loop labels in the next lesson!

Learn about loop labels in the next lesson!