

Slicing a Vector

This lesson teaches how to slice a vector in Rust!.

We'll cover the following ^

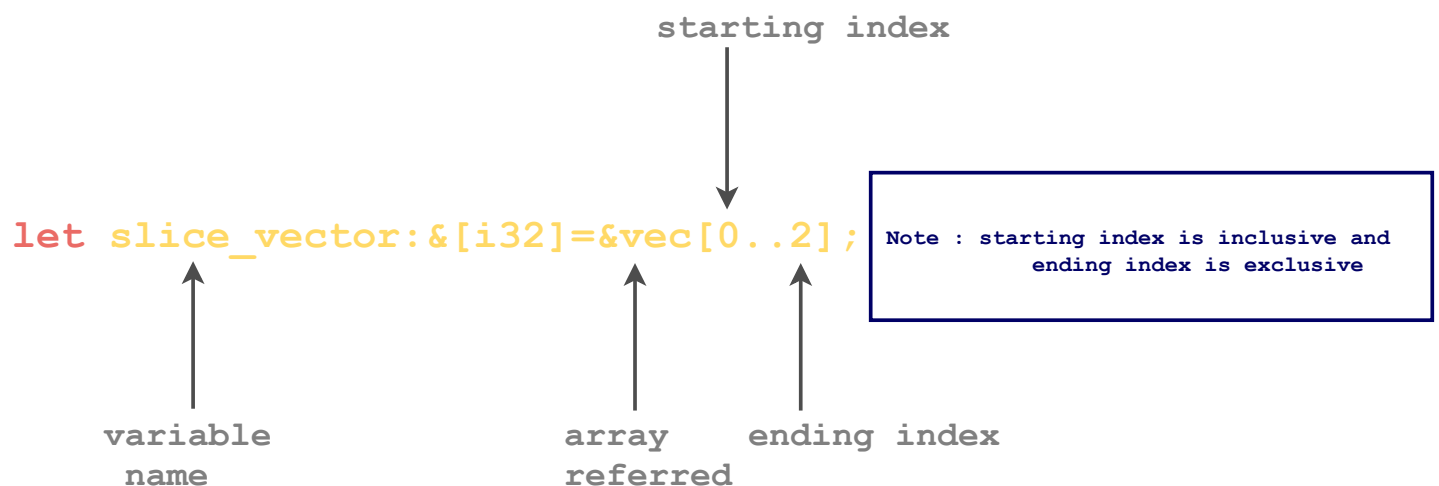
- Get Slice
 - Syntax
- Quiz

Get Slice

Imagine a situation where you need to get a portion of a vector. Rust allows you to borrow the slice of the vector instead of using the whole vector.

Syntax

Slice is a two-word object. The first word is a pointer to the data, and the second word is the length of the slice.



```
fn main() {  
    // define a vector of size 5  
    let my_vec = vec![1, 2, 3, 4, 5];  
    let slice:&[i32] = &my_vec[2..4];  
    // print the vector  
    println!("Slice of the vector : {:?}", slice);  
}
```

Quiz

Test your understanding of vector array slicing in Rust!

Quick Quiz on Slicing!

1

What is the output of the following code?

```
fn main() {  
    let my_vec = vec![1, 2, 3, 4, 5];  
    let slice:&[i32] = &my_vec[2..6];  
    println!("Slice of the vector : {:?}", slice);  
}
```

2

What is the output of the following code?

```
fn main() {  
    let my_vec = vec![2, 3, 9, 8, 7];  
    let slice:&[i32] = &my_vec[2..4];  
    println!("Slice of the vector : {:?}", slice);  
}
```

[Retake Quiz](#)

Now that you have learned about vectors in Rust, check your knowledge with a challenge in the next lesson.

