

变量、Mutability

- Rust默认变量是immutable的，除非在变量名前加一个 `mut`

```
fn main() {  
    let mut x = 5;  
    println!("The first value of x is {}", x);  
    x = 6;  
    println!("After changing the value of x, it has {}", x);  
}
```

- **Constant:**

- 与变量申明用 `let` 不同，constant前面是加 `const`. 并且constant是任何情况都immutable的
- 并且在申明const时要确定其数据类型
- const的声明周期是其所处的scope内

```
const THREE_HOURS_IN_SECONDS: u32 = 60 * 60 * 3;
```

- **Shadow**

- 特别要区分的是shadow和mut
- 看下面的代码：

```
let a = 5;  
let a = a + 1;  
{  
    let a = a * 2;  
    println!("The value of a rn is {}", a);  
}  
println!("The value of a out of scope is {}", a);
```

output:

```
The value of a rn is 12  
The value of a out of scope is 6
```

- shadow的实现是通过 `let` + 同样的变量名，新的值会遮盖前一个值，但当新的值out of scope的时候，旧的值又会重新出来。
- 并且对于 `mut a` 也是可以shadow的
- shadow遮盖的值数据类型甚至可以不同：

```
// shadow 2  
let letters = "abcde";  
{  
    let letters = letters.len();  
    println!("The value of letters rn is {}", letters);  
}  
println!("The value of letters out of scope is {}", letters);
```

output:

The value of letters rn is 5
The value of letters out of scope is abcde
