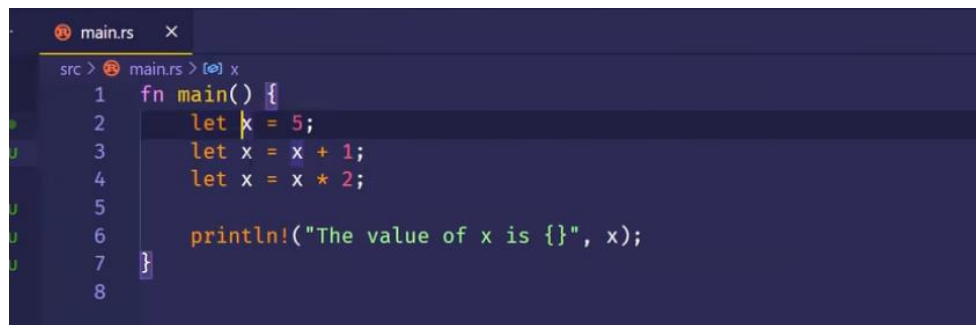


Shadow



```
main.rs x
src > main.rs > x
1 fn main() {
2     let x = 5;
3     let x = x + 1;
4     let x = x * 2;
5
6     println!("The value of x is {}", x);
7 }
8
```

用后定义的同名变量隐藏前面的变量

Shadowing（隐藏）

- 可以使用相同的名字声明新的变量，新的变量就会 shadow（隐藏）之前声明的同名变量
 - 在后续的代码中这个变量名代表的就是新的变量
 - （例子）
- shadow 和把变量标记为 mut 是不一样的：
 - 如果不使用 let 关键字，那么重新给非 mut 的变量赋值会导致编译时错误
 - 而使用 let 声明的同名新变量，也是不可变的
 - 使用 let 声明的同名新变量，它的类型可以与之前不同

简而言之就是如果使用 mut 定义变量的话 后续只能重新为变量赋值 不能改变它的类型 而用 shadow 相当于重新定义了一个新的变量 好处也是非常直接的