

1. 变量的命名（使用蛇形命名法）

```
let image_path = "images/img.jpg";           // 自动推导变量类型
let image_path: &str = "images/img.jpg";      // 手动声明变量类型
```

2. 变量的可变性

在 Rust 中变量默认是不可变的，除非使用 `mut` 关键字，`mut` 是 `mutable` 的缩写，代码在对应项目的 `main()` 中。

3. 下划线开头忽略未使用变量

代码在对应项目的 `underline()` 中。

4. 变量的解构和遮蔽（**shadowing**）

代码在对应项目的 `deconstruction()` 中。

5. 常量

- 常量不允许使用 `mut`。常量不仅仅默认不可变，而且自始至终不可变，因为常量在编译完成后，已经确定它的值。
- 常量使用 `const` 关键字而不是 `let` 关键字来声明，并且值的类型必须标注。
- 常量可以在任意作用域内声明，包括全局作用域，在声明的作用域内，常量在程序运行的整个过程中都有效。

```
const NAME: &str = "dream";
```