

# Раздел 3. Константы

В языке программирования Go, константы представляют собой именованные выражения, значения которых остаются неизменными на протяжении всего выполнения программы. Эти значения устанавливаются на этапе компиляции и не могут быть изменены во время выполнения программы. Они обычно используются для определения фиксированных значений, которые не должны изменяться, такие как число Пи, скорость света или любые другие специфические значения, которые неизменны по своей природе.

## Объявление констант

Вот пример объявления константы:

```
const Pi = 3.14159
```

В этом примере мы объявляем константу `Pi` и присваиваем ей значение `3.14159`. Константы могут быть объявлены с помощью ключевого слова `const`.

## Использование констант

После объявления константы мы можем использовать ее в программе. Например, мы можем использовать константу `Pi` для вычисления длины окружности:

```
const radius float64 = 5.0
const circumference = 2 * Pi * radius
```

В этом примере мы используем константу `Pi` для вычисления длины окружности. Мы объявляем переменную `radius` типа `float64` и присваиваем ей значение `5.0`. Затем мы используем эту переменную в вычислении длины окружности.

## Примеры использования констант

Вот несколько примеров использования констант в Go:

```
// Объявление константы типа string
const Greeting = "Hello, world!"

// Объявление константы типа int
const MaxInt = 2147483647

// Объявление константы типа float64
const Pi = 3.14159

// Использование констант в программе
fmt.Println(Greeting)
fmt.Println(MaxInt)
fmt.Println(Pi)
```

В этом примере мы объявляем константы `Greeting`, `MaxInt` и `Pi` и используем их в функции `Println` из библиотеки `fmt`, чтобы вывести значения констант на экран.

Примеры использования констант:

- Хранение значений математических констант, таких как `Pi` или `e`.
- Хранение значений, которые используются в качестве параметров в программе, например, максимальное число попыток входа в систему.
- Хранение значений, которые используются в качестве флагов, например, значения, которые указывают, включен или выключен режим отладки.

Константы могут быть объявлены в любом блоке кода, но обычно они объявляются в начале файла или пакета, чтобы обеспечить легкость чтения и поддержки кода.

Константы объявляются с помощью ключевого слова `const` и имеют имя, тип данных и значение.

В этом примере мы объявляем константу `Pi` и присваиваем ей значение `3.14159`.

## Шпаргалка

1. Константы - значения, которые не изменяются во время выполнения программы.
2. Хорошим тоном считается объявление констант в начале файла или пакета. Так код будет проще читать и поддерживать.
3. Константы, как и переменные, могут быть разных типов. Например `int`, `float64`, `string` и `bool`.
4. Константы, как и переменные, могут быть определены как явно, так и неявно.
5. Разница между константой и переменной заключается в том, что константы объявляются с помощью ключевого слова `const` и не могут менять своё значение во время выполнения программы. Переменные - именованные участки памяти, объявляются ключевым словом `var` и могут изменять значение.

## Домашнее задание

1. **Объявление глобальных констант:** Объявите глобальную константу и используйте ее внутри функции. Выведите значение этой константы.
2. **Объявление локальных констант:** Внутри функции объявите локальную константу и используйте ее. Выведите значение этой константы.
3. **Сравнение глобальных и локальных констант:** Объявите глобальную и локальную константы с одинаковыми именами, но разными значениями. Выведите обе константы внутри функции. Объясните полученный результат.
4. **Использование констант в математических операциях:** Объявите две константы, присвойте им числовые значения. Напишите программу, которая выводит результат сложения, вычитания, умножения и деления этих констант.
5. **Неизменяемость констант:** Попробуйте изменить значение константы после ее объявления. Опишите полученный результат и объясните, почему он такой.
6. **Типы данных констант:** Объявите константы различных типов (`int`, `float64`, `string`, `bool`) и используйте их в программе. Выведите значения этих

констант.

7. **Математические операции с константами и переменными:** Создайте переменную и константу с типами `int`, присвойте им значения и выполните различные математические операции. Проведите аналогичные операции для типа `float64`
8. **Конкатенация констант и переменных:** Создайте переменную и константу с типами `string`, присвойте им значения и выведите результат их конкатенации.