

Раздел 7. Импорты

В языке программирования Go импорты используются для включения кода из других пакетов в текущую программу. Импортируемый код может содержать функции, типы данных, константы и многое другое.

Для импорта пакетов в Go используется ключевое слово `import`, после которого указывается путь к необходимому пакету. Посмотрим на пример уже знакомого нам импорта:

```
import "fmt"
```

Этот импорт включает код из пакета `fmt`, который содержит функции для форматирования и вывода данных.

Также есть возможность импортировать пакеты, используя псевдонимы, это делает код более читаемым. Взглянем на пример:

```
import (  
    "fmt"  
    log "myproject/logger"  
)
```

В данном случае, мы импортируем стандартный пакет `fmt` и пакет `logger` из проекта `myproject`, используя псевдоним `log`.

С версии Go 1.11, в языке появилась поддержка модулей, что позволило улучшить управление зависимостями. При использовании модулей, вы можете создать файл `go.mod` в корне вашего проекта, и Go автоматически определит, где находятся необходимые пакеты.

В Go есть возможность импортировать только отдельные элементы из пакета, указав их имя. Например:

```
import "fmt"  
  
func main() {
```

```
    fmt.Println("Hello, world!")
}
```

В этом примере мы импортировали только функцию `Println` из пакета `fmt` и использовали ее для вывода текста.

Стандартная библиотека Go предлагает множество полезных пакетов.

Например, пакет `strconv` предоставляет функции для преобразования строк в другие типы данных:

```
import "strconv"

func main() {
    i, _ := strconv.Atoi("10")
    fmt.Println(i + 5) // Выводит 15
}
```

В этом примере мы использовали функцию `Atoi` из пакета `strconv` для преобразования строки `"10"` в целое число.

Шпаргалки

1. Импорты в Go - это мощный инструмент для включения кода из других пакетов в программу. Они позволяют программистам использовать готовые решения, создавать модульный код и повторно использовать код в различных проектах.

Домашнее задание

1. **Использование стандартной библиотеки:** Импортируйте пакет `"math"` и используйте его для вычисления квадратного корня числа.
2. **Использование нескольких пакетов:** Импортируйте пакеты `"math"` и `"time"`, используйте `"math"` для вычисления квадратного корня текущего времени в секундах.
3. **Использование псевдонимов для пакетов:** Импортируйте пакет `"math/rand"` с псевдонимом и используйте его для генерации случайного числа.

4. **Использование специфической функции из пакета:** Импортируйте только функцию `Sqrt` из пакета `"math"` и используйте ее в вашей программе.