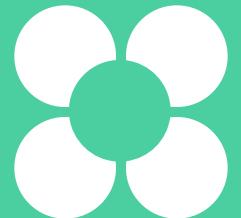


# Среда разработки и запуск программ

Игорь Шамаев

Руководитель направления разработки



# Шамаев Игорь

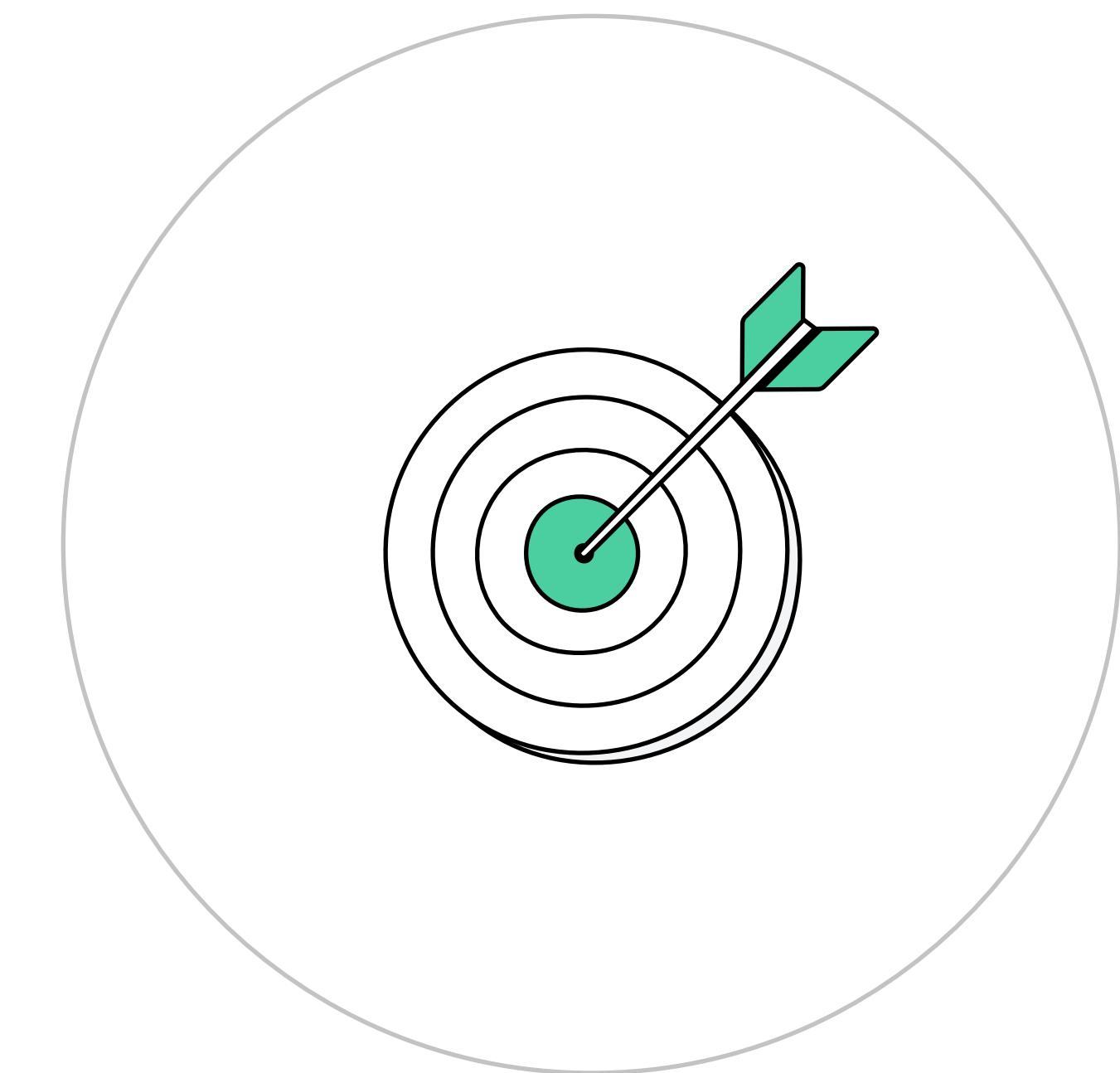
О спикере:

- Руководитель направления разработки в ДомКлик
- Развивает и поддерживает внутреннюю базу данных управления конфигурацией на Golang и TypeScript
- В прошлом – руководитель команды разработки в Сбере



# Цели занятия

- Узнать, как установить и настроить среду разработки для языка Go на своем компьютере
- Познакомиться с основными компонентами среды разработки и их назначением
- Разобраться, как компилировать и запускать программы на языке Go
- Понять, как диагностировать и исправлять ошибки, возникающие при компиляции и выполнении программ



# План занятия

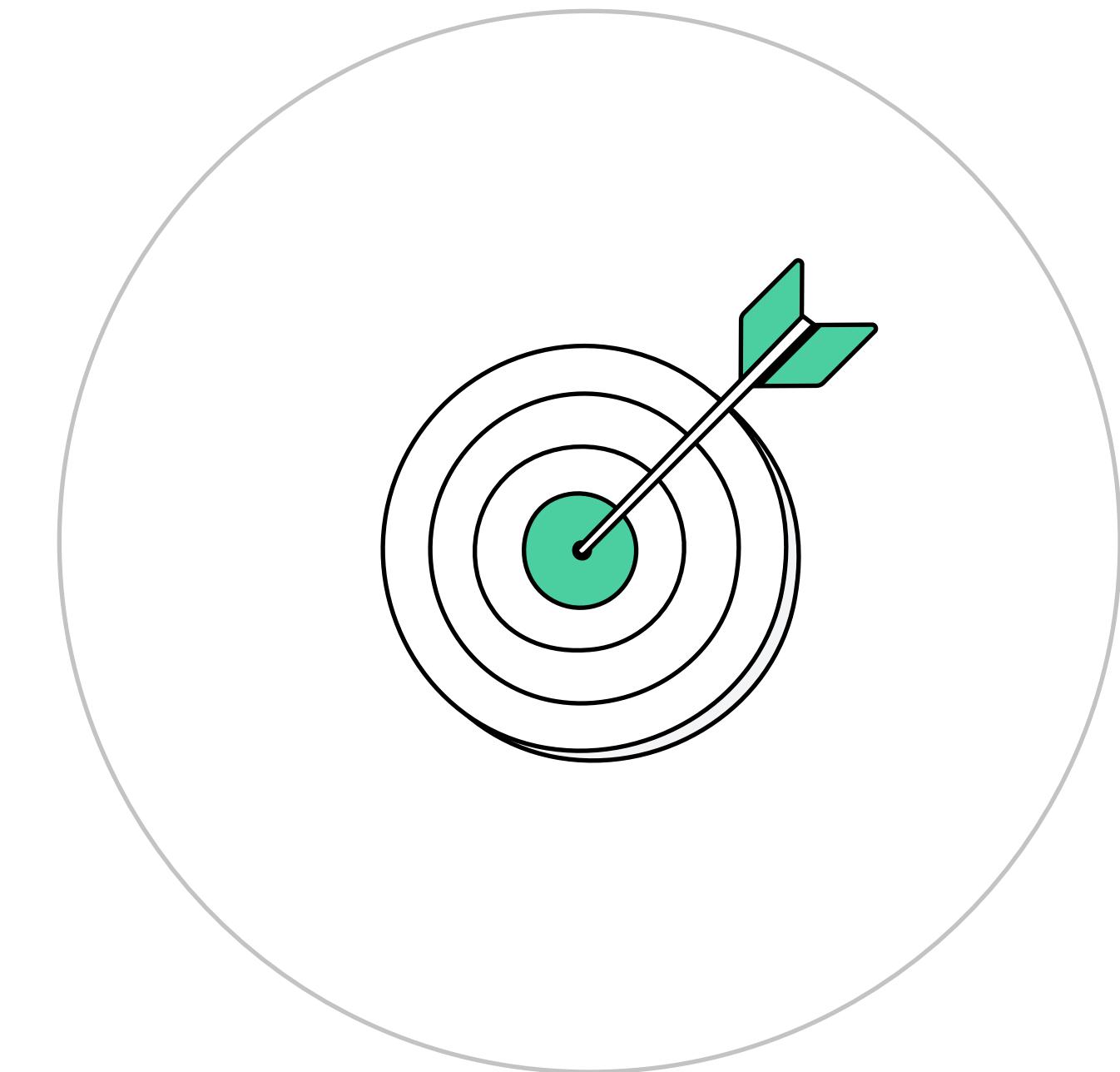
- 1 Установка и настройка среды разработки
- 2 Основные компоненты среды разработки и их назначение
- 3 Компиляция и запуск программ
- 4 Диагностика и исправление ошибок



# Установка и настройка среды разработки

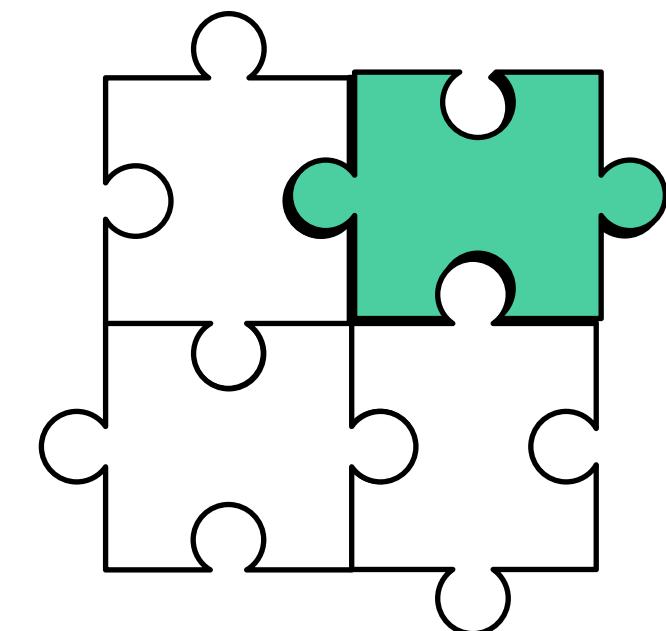
# Цели видео

- Ознакомиться с официальными источниками загрузки и выбором подходящего установщика для своей ОС
- Разобрать процесс установки и узнать особенности установки на различных операционных системах
- Научиться запускать команды проверки для подтверждения успешной установки Go
- Настроить среду разработки для работы с Go



# Процесс установки

- 1 Скачать установщик с официального сайта [go.dev/dl](https://go.dev/dl)
- 2 Запустить установку
- 3 Проверить установку



# Загрузка установочного файла

Перейдите на страницу загрузки и скачайте  
установочный файл

## Microsoft Windows

Windows 10 or later, Intel 64-bit processor

 [go1.24.2.windows-amd64.msi](#)

## Apple macOS (ARM64)

macOS 11 or later, Apple 64-bit processor

 [go1.24.2.darwin-arm64.pkg](#)

## Apple macOS (x86-64)

macOS 11 or later, Intel 64-bit processor

 [go1.24.2.darwin-amd64.pkg](#)

## Linux

Linux 2.6.32 or later, Intel 64-bit processor

 [go1.24.2.linux-amd64.tar.gz](#)

## Source

 [go1.24.2.src.tar.gz](#)

# Установка на Microsoft Windows и Apple macOS

- Откройте папку с загруженным MSI-файлом
- Запустите файл двойным кликом мыши
- Следуйте инструкциям мастера установки

The screenshot shows a dark-themed web page with tabs for Linux, Mac, and Windows. The Windows tab is selected. Step 1 instructs users to open the MSI file and follow prompts to install Go. It notes that by default, the installer will install Go to Program Files or Program Files (x86). Step 2 instructs users to verify installation by opening the Start menu, searching for cmd, and running go version in the Command Prompt window. A code block shows the command \$ go version being typed.

1. Open the [MSI file](#) you downloaded and follow the prompts to [install Go](#).  
By default, the installer will install Go to [Program Files](#) or [Program Files \(x86\)](#). You can change the location as needed. After installing, you will need to close and reopen any open command prompts so that changes to the environment made by the installer are reflected at the command prompt.

2. Verify that you've [installed Go](#).

1. In **Windows**, click the **Start** menu.
2. In the menu's search box, type `cmd`, then press the **Enter** key.
3. In the Command Prompt window that appears, type the following command:

```
$ go version
```

4. Confirm that the command prints the installed version of Go.

# Установка на Linux

- Удалите все предыдущие установки Go
- Добавьте `/usr/local/go/bin` в переменную окружения PATH
- Убедитесь, что вы установили Go, открыв командную строку и введя команду

The screenshot shows a dark-themed web page with tabs for Linux, Mac, and Windows. The Linux tab is selected. Step 1 details the removal of previous Go installations and extraction of a new archive. Step 2 covers adding the Go bin directory to the PATH environment variable. Step 3 involves verifying the installation by running the command `$ go version`. Step 4 is a placeholder for confirming the installed version.

Linux   Mac   Windows

1. Remove any previous Go installation by deleting the `/usr/local/go` folder (if it exists), then extract the archive you just downloaded into `/usr/local`, creating a fresh Go tree in `/usr/local/go`:

```
$ rm -rf /usr/local/go && tar -C /usr/local -xzf go1.24.2.linux-amd64.tar.gz
```

(You may need to run the command as root or through sudo).

**Do not** untar the archive into an existing `/usr/local/go` tree. This is known to produce broken Go installations.

2. Add `/usr/local/go/bin` to the PATH environment variable.

You can do this by adding the following line to your `$HOME/.profile` or `/etc/profile` (for a system-wide installation):

```
export PATH=$PATH:/usr/local/go/bin
```

**Note:** Changes made to a profile file may not apply until the next time you log into your computer. To apply the changes immediately, just run the shell commands directly or execute them from the profile using a command such as `source $HOME/.profile`.

3. Verify that you've installed Go by opening a command prompt and typing the following command:

```
$ go version
```

4. Confirm that the command prints the installed version of Go.

# Проверка установки

Откройте командную строку и введите следующую команду

```
$ go version
```

# Ошибка выполнения команды

Добавляем путь в переменную окружения PATH:

- для Apple macOS версии равной и выше Catalina (10.15)

```
echo 'export PATH=$PATH:/usr/local/go/bin' >> ~/.zshrc, закрыть и заново  
открыть терминал
```

- для Linux и Apple macOS версии меньше Catalina (10.15)

```
'export PATH=$PATH:/usr/local/go/bin' >> ~/.bashrc, закрыть и заново  
открыть терминал
```

# Ошибка выполнения команды

Добавляем путь в переменную окружения PATH:

- для Microsoft Windows

```
setx Path "%Path%;C:\Go\bin" /M из терминала от имени администратора,  
закрыть и заново открыть терминал
```

# Процесс настройки

- 1 Установить Visual Studio Code
- 2 Добавить расширение Go

# Загрузка установочного файла Visual Studio Code

Перейдите на страницу загрузки и скачайте установочный файл



**↓ Windows**  
Windows 10, 11

**↓ .deb**  
Debian, Ubuntu

**↓ .rpm**  
Red Hat, Fedora, SUSE

**↓ Mac**  
macOS 10.15+

User Installer [x64](#) [Arm64](#)

System  
Installer [x64](#) [Arm64](#)

[.zip](#) [x64](#) [Arm64](#)

CLI [x64](#) [Arm64](#)

.deb [x64](#) [Arm32](#) [Arm64](#)

.rpm [x64](#) [Arm32](#) [Arm64](#)

.tar.gz [x64](#) [Arm32](#) [Arm64](#)

Snap [Snap Store](#)

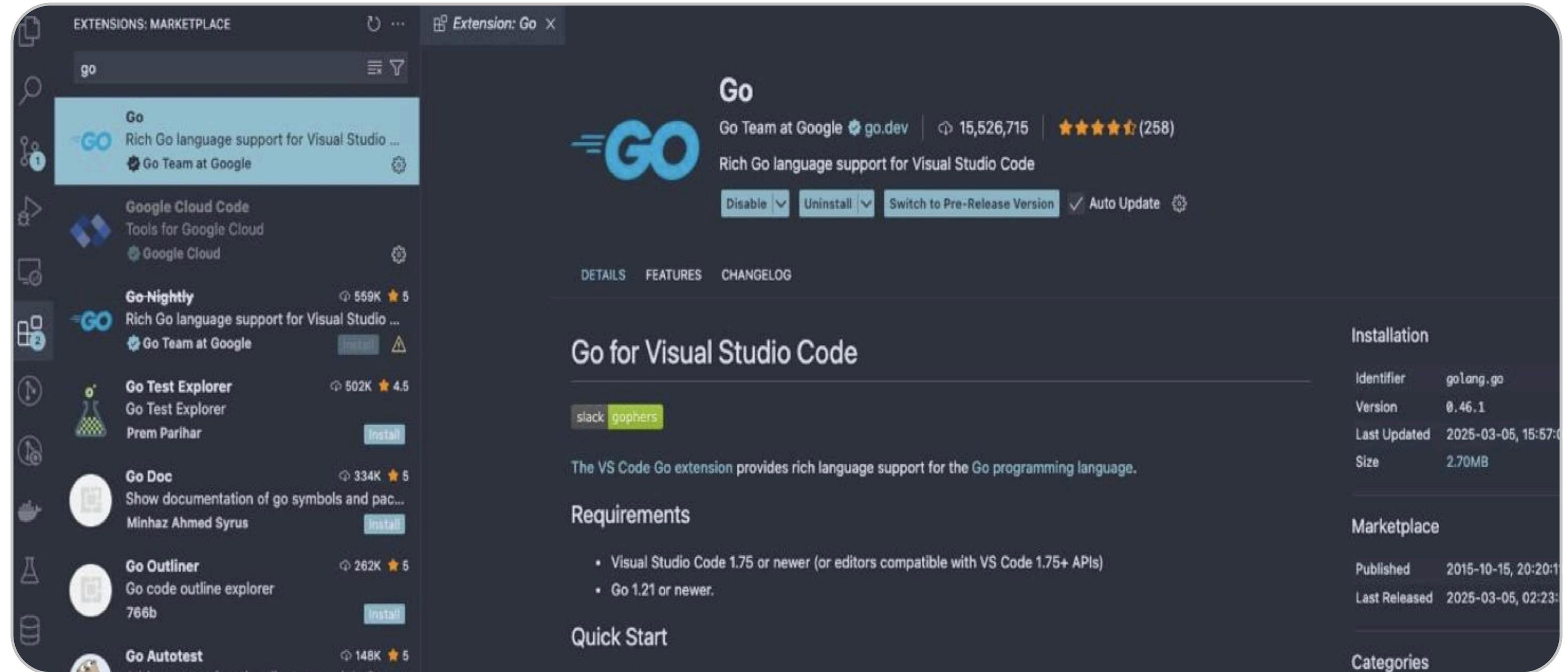
CLI [x64](#) [Arm32](#) [Arm64](#)

.zip [Intel chip](#) [Apple silicon](#) [Universal](#)

CLI [Intel chip](#) [Apple silicon](#)

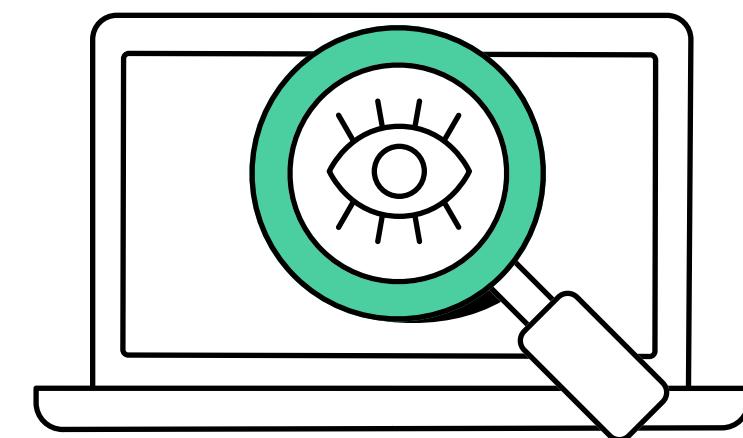
# Установка расширения для Go

- Откройте магазин VS Code
- Введите в поиск Go
- Нажмите установить

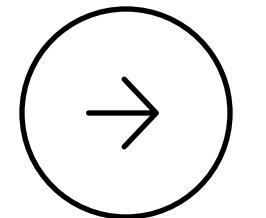


# Демонстрация работы

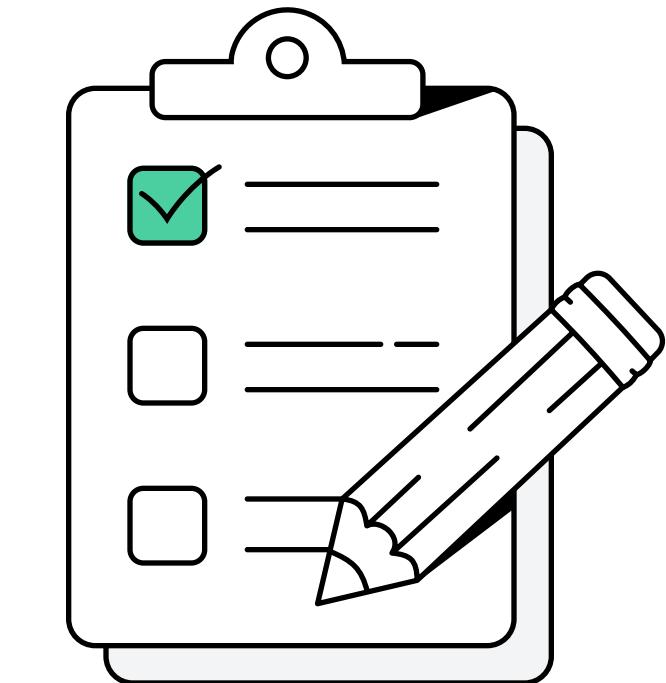
Скачивание и установка программ



# Итоги



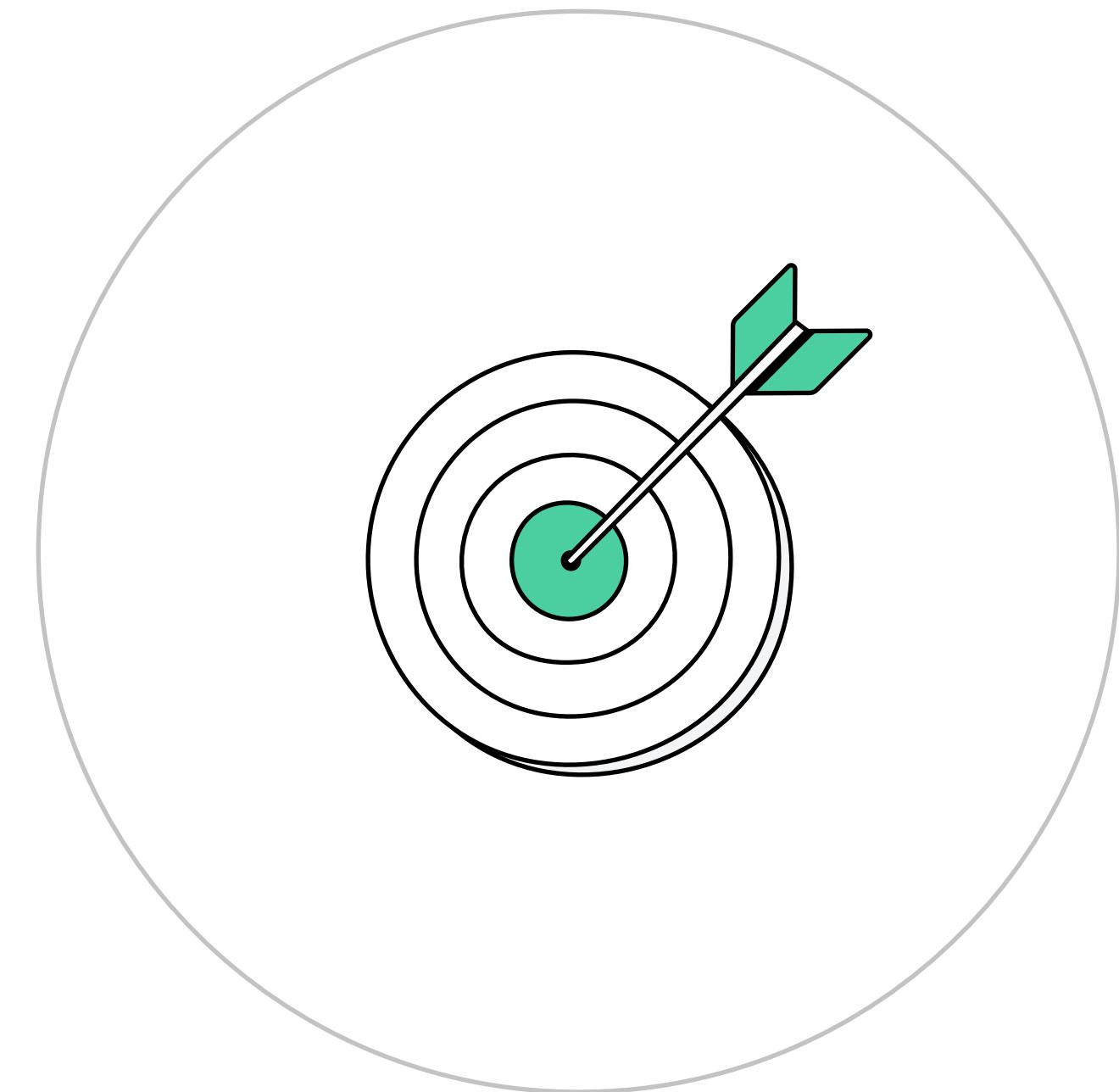
Установили и настроили среду для работы с Go,  
включая выбор установщика и проверку установки  
на различных ОС



# Основные компоненты среды разработки и их назначение

# Цели видео

- Изучить основные компоненты среды разработки



# Компоненты среды разработки

- Go
- Visual Studio Code
- Расширение Visual Studio Code для работы с Go

# Назначение Go

Предоставляет компилятор, стандартную библиотеку и инструменты для разработки

Функции:

- Компиляция и запуск программ
- Управление зависимостями
- Тестирование, форматирование кода, статический анализ

# Назначение Visual Studio Code

Редактор кода

Функции:

- Подсветка синтаксиса, автодополнение, навигация по коду
- Поддержка отладки и запуска программ
- Гибкость за счёт расширений

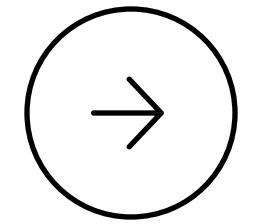
# **Назначение расширения Visual Studio Code для Go**

Автоматическая проверка кода в редакторе

**Функции:**

- Автодополнение кода
- Навигация по коду
- Форматирование кода
- Всплывающие подсказки

# Итоги



Разобрали ключевые элементы среды разработки  
и их назначение



# Заключение

# Итоги занятия

- Установили Go на свою ОС (Windows, macOS или Linux)
- Настроили VS Code для работы с Go, включая плагины
- Разобрались, как компилировать и запускать программы на языке Go
- Изучили диагностику и исправление ошибок, возникающих при компиляции и выполнении программ



# Среда разработки и запуск программ

Игорь Шамаев

Руководитель направления разработки

