



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное  
учреждение высшего образования  
«Московский государственный технический университет имени  
Н. Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н. Э. Баумана)

---

ФАКУЛЬТЕТ «Информатика и системы управления»

---

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

---

## Мануал по работе с ClosureScript

# Содержание

Введение	3
Список использованных источников	9

# Введение

## **Что такое ClosureScript?**

ClosureScript [1] — компилятор для Closure [2], выдающий в результате код на JavaScript.

## **Что такое Closure?**

Closure — диалект языка LISP, являющийся динамическим компилируемым языком программирования, поддерживающий доступ к фреймворкам, написанным на Java. Из-за своего родства с LISP поддерживает функциональное программирование и использование макросов.

## **Что нужно для того, чтобы начать писать на Closure?**

Во-первых, среда разработки или текстовый редактор для Closure — подходящих несколько, например Emacs, IntelliJ IDEA, VS Code. В рамках данного мануала будет рассмотрено использование текстового редактора VS Code для работы с Closure.

Во-вторых, сам Closure — он доступен для установки под MacOS, Linux и Windows.

В-третьих,

# Установка

## VS Code и Calva

Установить VS Code под свою платформу можно по ссылке: <https://code.visualstudio.com/Download>

Для тех, кто не имеет опыта использования IDE или желает научиться использовать VS Code, рекомендуются к прочтению следующие статьи:

- 1) <https://habr.com/ru/post/490754/> — статья на русском языке
- 2) <https://code.visualstudio.com/docs/introvideos/basics> — статья на английском языке

Далее необходимо установить Calva — расширение для VS Code, поддерживающее ClojureScript и помогающее разрабатывать ПО на Clojure. Для этого в левом меню VS Code необходимо перейти в раздел "Расширения" (см. скриншот 1), сделать поиск "Calva" и установить найденное расширение (см. скриншот 2).

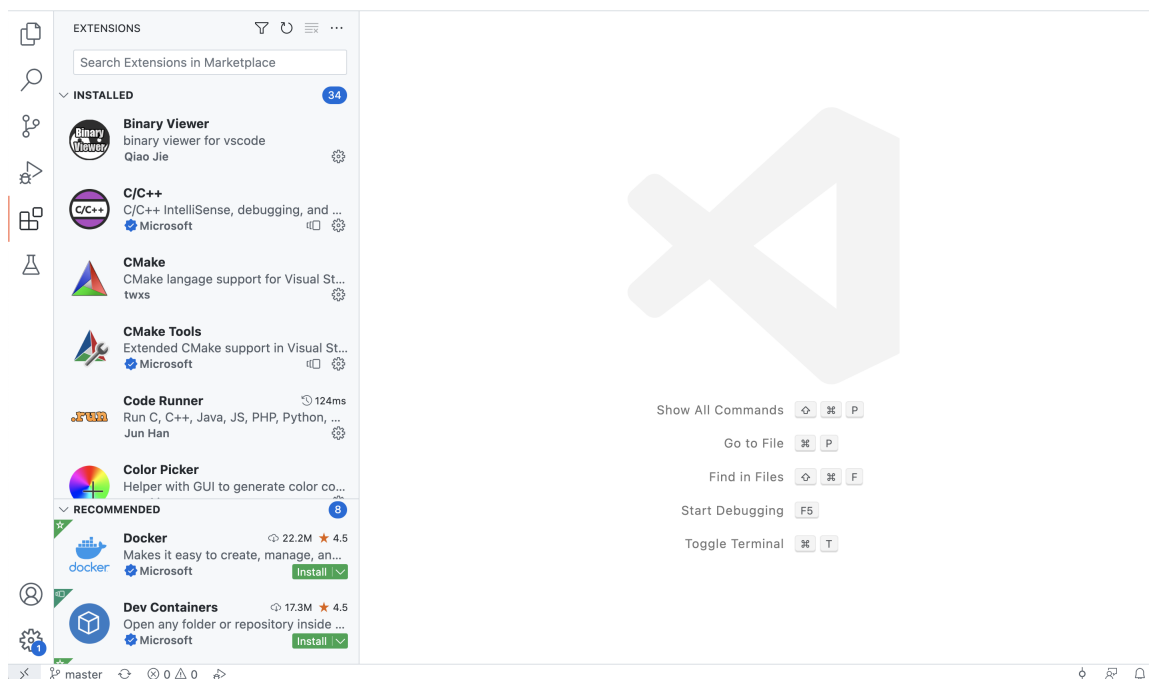


Рисунок 1 – Раздел "Расширения" VS Code

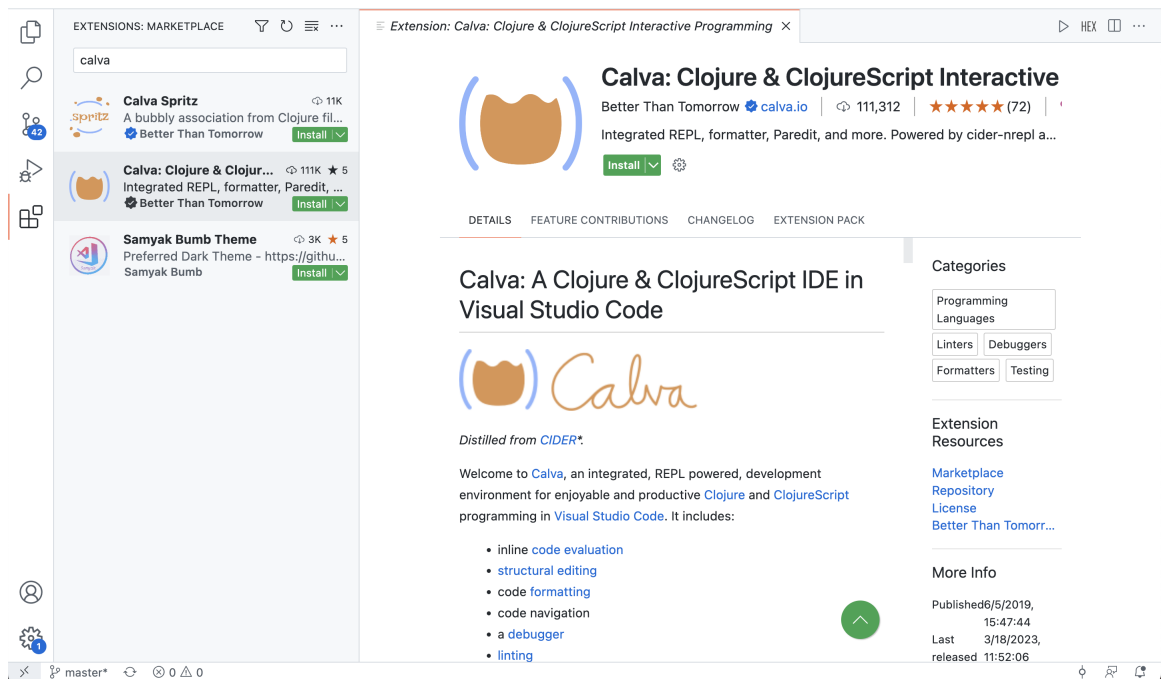


Рисунок 2 – Расширение Calva

## Clojure

Для того, чтобы установить Clojure, воспользуемся инструкциями с официального сайта: [https://clojure.org/guides/install\\_clojure](https://clojure.org/guides/install_clojure).

В приведённой статье (на английском языке) присутствуют инструкции по установке Clojure на ОС MacOS, Linux-подобные ОС (Ubuntu, Debian), а также Windows. В данном мануале будет рассмотрена установка Clojure на Linux-подобные ОС посредством установщика пакетов brew.

Для установки необходимо открыть терминал и выполнить в нём следующие команды:

### Листинг 1 – Установка brew

```
1 /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
2 (echo; echo 'eval "$( /home/linuxbrew/.linuxbrew/bin/brew shellenv )"'
3   ) >> /home/<YOURUSERNAME>/.profile
4 eval "$( /home/linuxbrew/.linuxbrew/bin/brew shellenv )"
5 sudo apt-get install build-essentials
```

После установки brew необходимо установить и сам Clojure:

### Листинг 2 – Установка Clojure

```
1 brew install clojure/tools/clojure
```

## LISP

Поскольку Clojure является диалектом LISP, стоит также установить и базовый интерпретатор LISP, например, Common Lisp. Для ОС семейства Linux это можно сделать следующей командой:

### Листинг 3 – Установка Common Lisp

```
1 sudo apt-get install sbcl
```

# Начало работы и примеры

## Hello World, Clojure

Откроем VS Code и нажмём Ctrl + Shift + P (данное сочетание клавиш откроет паллет управления в VS Code) и введём следующий текст: Calva: Fire up the Getting Started REPL Среди появившихся результатов выберем единственный:

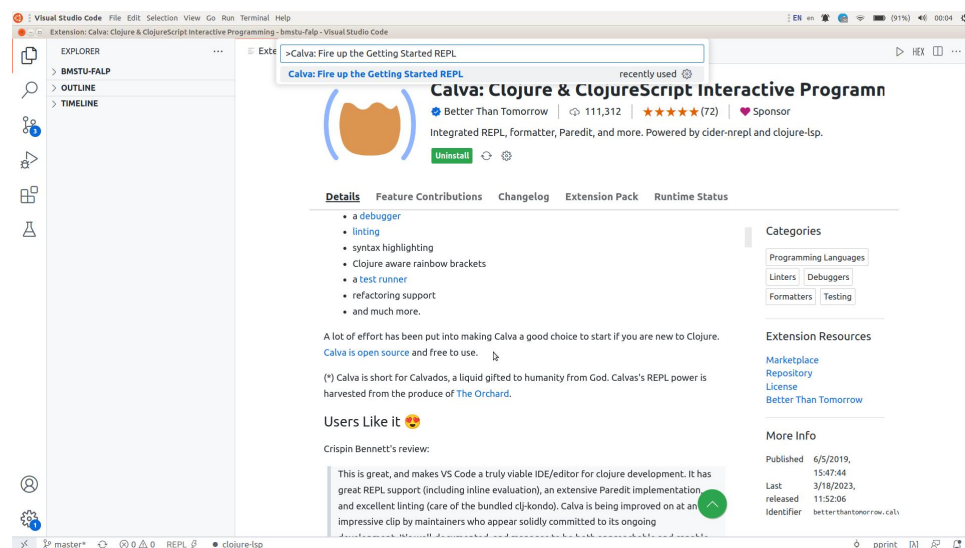


Рисунок 3 – Запуск Calva

После запуска Calva мы увидим примерно следующее:

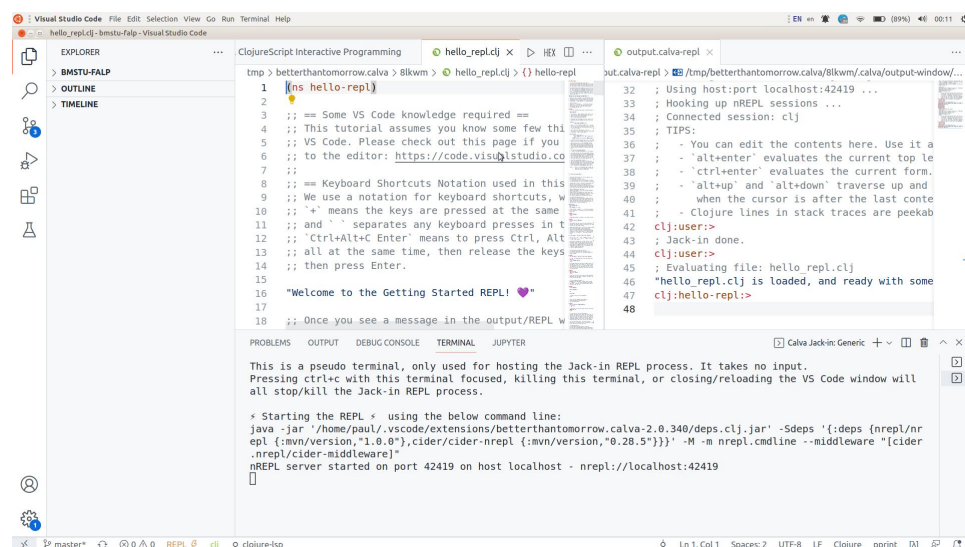


Рисунок 4 – После запуска Calva

Очистим содержимое левого файла и напомним в него "Hello, World!":

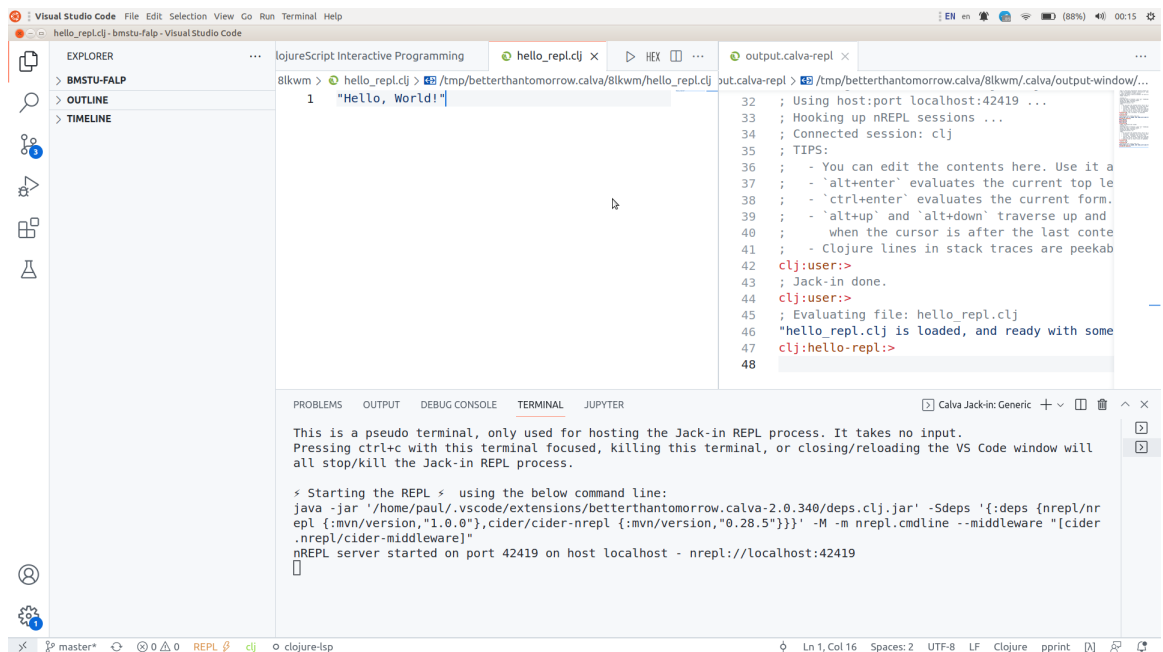


Рисунок 5 – Напишем "Hello, World!"

Нажмём Alt + Enter и увидим результат:

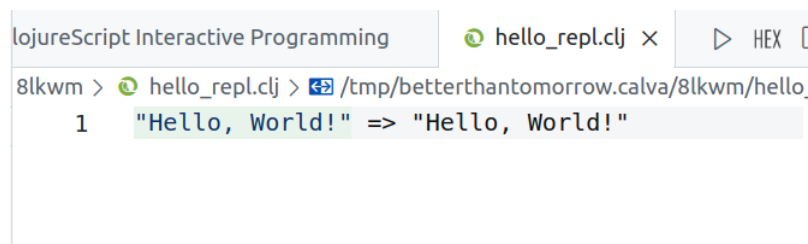


Рисунок 6 – Hello, World!

# Hello World, ClojureScript



# Список использованных источников

1. ClojureScript [Электронный ресурс]. Режим доступа: <https://clojurescript.org/> (дата обращения: 22.03.2023).
2. Closure [Электронный ресурс]. Режим доступа: <https://closure.org/> (дата обращения: 22.03.2023).
3. Баварин И. И. Высшая математика: учебник по естественно-научным направлениям и специальностям. – М.: Гуманит. изд. центр ВЛАДОС, 2003. С. 45–49.
4. Kakaradov B. Ultra-Fast Matrix Multiplication: An Empirical Analysis of Highly Optimized Vector Algorithms, 2004 [Электронный ресурс]. Режим доступа: [https://cs.stanford.edu/people/boyko/pubs/MatrixMult\\_SURJ\\_2004.pdf](https://cs.stanford.edu/people/boyko/pubs/MatrixMult_SURJ_2004.pdf) (дата обращения: 04.10.2022).