



LISP

УПРАВЛЕНИЕ СЛОЖНОСТЬЮ КОДА И ОБОБЩЕНИЯ

СТУДЕНТЫ ГРУППЫ 5030102/20202:
ГЕОРГИЙ ШТЕЙНБЕРГ
СЕРГЕЙ ТИШКОВЕЦ

ПЛАН ПРЕЗЕНТАЦИИ

- Система пакетов
- Пространства имен и экспорт
- Обобщенные функции
- Системы – ASDF
- Менеджер библиотек – Quicklisp
- Введение в ООП
- Список литературы

СИСТЕМА ПАКЕТОВ И ПРОСТРАНСТВА ИМЕН

Проблема: Как организовать большой проект?

Решение: Система пакетов (Packages) и пространства имен

Пакет - это контейнер для символов, предотвращающий конфликты имен

- ***defpackage*** - определение пакета
- ***in-package*** - переключение в пакет
- ***use-package*** - использование другого пакета

Пространства имен – символы квалифицируются пакетом

- **локальные/экспортируемые**

```
10  ✓ (defpackage :my-package
11      (:use :common-lisp)
12      (:export :my-function :my-variable))
13
14      (in-package :my-package)
15
16      (defvar my-variable 42)
17
18  ✓ (defun my-function (x)
19      (* x 2))
20
21      (print my-variable) ; Вывод: 42
22      (print (my-function 5)) ; Вывод: 10
23
24      (in-package :cl-user)
25
26      (use-package :my-package)
27
28      (print my-variable) ; Вывод: 42
29      (print (my-function 5)) ; Вывод: 10
```

ОБОБЩЕННЫЕ ФУНКЦИИ (GENERIC FUNCTIONS)

Обобщенные функции - работают с разными типами данных

Множественная диспетчеризация - выбор реализации на основе всех аргументов

Отличие от ООП: В ООП метод принадлежит классу, в Lisp - функция независима

- ***defgeneric*** – определение обобщенной функции
- ***defmethod*** – добавление методов для конкретных типов

```
34 (defgeneric area (shape &key &allow-other-keys)
35   (:documentation "Обобщенная функция для вычисления площади фигуры"))
36
37 (defmethod area ((shape (eql :circle)) &key (radius 0))
38   (* pi radius radius)) ; Метод для круга
39
40 (defmethod area ((shape (eql :rectangle)) &key (length 0) (width 0))
41   (* length width)) ; Метод для прямоугольника
42
43 (print (area :circle :radius 5)) ; Вывод: примерно 78.5398
44 (print (area :rectangle :length 4 :width 6)) ; Вывод: 24
```

МЕНЕДЖЕР БИБЛИОТЕК - QUICKLISP

- ***(ql:quickload "library-name")*** – установка и загрузка библиотеки
- ***(ql:system-*apropos* "keyword")*** – поиск библиотек
- ***(ql:update-all-dists)*** – обновление репозиториев

```
0] (ql:quickload "cl-ppcre")

"Updated Quicklisp distributions""Updated Quicklisp distributions"
0] (ql:system-apropos "ppcre")
#<SYSTEM arnesi/cl-ppcre-extras / arnesi-20250622-git / quicklisp 2025-06-22>
#<SYSTEM cl-ppcre / cl-ppcre-20250622-git / quicklisp 2025-06-22>
#<SYSTEM cl-ppcre-template / cl-unification-20250622-git / quicklisp 2025-06-22>
#<SYSTEM cl-ppcre-unicode / cl-ppcre-20250622-git / quicklisp 2025-06-22>
#<SYSTEM cl-ppcre-unicode/test / cl-ppcre-20250622-git / quicklisp 2025-06-22>
#<SYSTEM cl-ppcre/test / cl-ppcre-20250622-git / quicklisp 2025-06-22>
#<SYSTEM optima.ppcr / optima-20150709-git / quicklisp 2025-06-22>
#<SYSTEM parser-combinators-cl-ppcre / cl-parser-combinators-20131111-git / quicklisp 2025-06-22>
#<SYSTEM trivia.ppcr / trivia-20241012-git / quicklisp 2025-06-22>

0] (ql:update-all-dists)
1 dist to check.
You already have the latest version of "quicklisp": 2025-06-22.
```

СИСТЕМЫ - ASDF

ASDF (Another System Definition Facility) – стандарт для определения и сборки проектов

- Компиляция и загрузка компонентов в правильном порядке
- Управление зависимостями
- Тестирование и распространение

Пример установки и использования ASDF:

- Создать файлы: my-system.asd, package.lisp, main.lisp.
- Установить Quicklisp:

```
C:\LISP\projectLisp>curl -O https://beta.quicklisp.org/quicklisp.lisp
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total     Spent    Left  Speed
100 57144  100 57144    0     0  462k      0 --:--:-- --:--:-- --:--:-- 468k
C:\LISP\projectLisp>docker run -it --rm -v %CD%:/project daewok/lisp-devel:base sbcl
* (load "/project/quicklisp.lisp")
* (quicklisp-quickstart:install)
* (ql:add-to-init-file)
* (ql:quickload "cl-ppcre")
```

СИСТЕМЫ - ASDF

Содержимое файлов my-system.asd, package.lisp, main.lisp

```
my-system.asd – Блокнот
Файл Правка Формат Вид Справка
(asdf:defsystem "my-system"
 :description "My Lisp system"
 :author "Your Name"
 :version "0.1"
 :components ((:file "package")
              (:file "main" :depends-on ("package")))
 :depends-on ("cl-ppcre"))
```

```
C: > LISP > projectLisp > package.lisp
1 (defpackage :my-system
2   (:use :cl)
3   (:export :hello))
```

```
47 (in-package :my-system)
48
49 (defun hello ()
50   (format t "Hello from my-system!~%"))
```

Выполнение команды:

```
* (require :asdf)
(asdf:load-asd "/project/my-system.asd")
(ql:quickload "cl-ppcre")
(asdf:load-system "my-system")
(my-system:hello)NIL
```

Вывод:

```
Hello from my-system!
NIL
*
```

ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

CLOS (Common Lisp Object System) — мощная ООП-система, интегрированная в Lisp

- Множественное наследование
- Обобщенные функции для методов
- Метаобъектный протокол (MOP)
- Методы-комбинации — определяют порядок вызова методов:
 - **:before** — выполняется до основного метода
 - **:after** — выполняется после основного метода
 - **:around** — оборачивает вызов метода

ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

```
79 (defclass shape ()
80 | ((color :initarg :color :accessor shape-color :initform "red")))
81
82 (defclass circle (shape)
83 | ((radius :initarg :radius :accessor circle-radius)))
84
85 (defgeneric draw (shape)
86 | (:documentation "Обобщенная функция для рисования фигуры"))
87
88 (defmethod draw ((s shape))
89 | (print (format nil "Drawing shape with color ~a" (shape-color s))))
90
91 (defmethod draw :before ((s circle))
92 | (print (format nil "Before drawing circle with radius ~a" (circle-radius s))))
93
94 (defmethod draw :after ((s circle))
95 | (print "After drawing circle"))
96
97 (defmethod draw :around ((s circle))
98 | (print "Around drawing circle")
99 | (call-next-method)
100 | (print "End around drawing circle"))
101
102 (let ((c (make-instance 'circle :radius 5 :color "blue")))
103 | (draw c)) ; ВЫВОД:
104 |         ; "Around drawing circle"
105 |         ; "Before drawing circle with radius 5"
106 |         ; "Drawing shape with color blue"
107 |         ; "After drawing circle"
108 |         ; "End around drawing circle"
```



СПИСОК ЛИТЕРАТУРЫ

- [LISP - Краткое руководство](#)
- [Основы программирования на языке Lisp](#)

Abstract geometric lines in white on a black background, forming various polygons and intersecting lines on the left side of the slide.

СПАСИБО ЗА ВНИМАНИЕ