



LISP

ОСНОВЫ ДАННЫХ И ТИПОВ

СТУДЕНТЫ ГРУППЫ 5030102/20202:
ГЕОРГИЙ ШТЕЙНБЕРГ
СЕРГЕЙ ТИШКОВЕЦ

ПЛАН ПРЕЗЕНТАЦИИ

- Типы данных в Lisp
- Операции над типами
- Преобразование/приведение типов
- Сравнения типов
- Список литературы

ТИПЫ ДАННЫХ: ПРИМИТИВНЫЕ

- **Числа (Number):**
 - **integer** – целые числа
 - **float** – числа с плавающей точкой
 - **ratio** – рациональные дроби
- **Символ (Symbol)** – уникальное имя, часто используемое как идентификатор переменной или функции
- **Строка (String)** – последовательность символов в двойных кавычках
- **Булевы значения (Boolean):**
 - **t** – true
 - **nil** – false
- **Nil** – пустой список ()
- **Cons-ячейка (Cons Cell)** – фундаментальная строительная единица списков. Представляет собой пару из двух указателей: **car** и **cdr**

ТИПЫ ДАННЫХ: ПОЛЬЗОВАТЕЛЬСКИЕ

1. Структуры (Struct) – создаются макросом **defstruct**

```
44 ;; Структуры (Struct)
45 (defstruct person
46   name
47   age)
48
49 (defvar ivan (make-person :name "Ivan" :age 30)) ; Создание экземпляра структуры
50
51 (print (person-name ivan)) ; Вывод: "Ivan"
52 (print (person-age ivan)) ; Вывод: 30
```

2. Классы (CLOS) – создаются макросом **defclass**

```
54 ;; Классы (CLOS)
55 (defclass animal ()
56   ((species :initarg :species :accessor species)
57    (sound :initarg :sound :accessor sound)))
58
59 (defvar cow (make-instance 'animal :species "Cow" :sound "Muuu"))
60
61 (print (species cow)) ; Вывод: "Cow"
62 (print (sound cow)) ; Вывод: "Muuu"
```

ОПЕРАЦИИ НАД ТИПАМИ

- **(type-of object)**
 - Возвращает тип переданного объекта
 - Пример: **(type-of 5) -> integer**
- **(typep object type)**
 - Проверяет, принадлежит ли объект указанному типу
 - Возвращает *t* или *nil*
 - Пример: **(typep 100 'integer) -> t**

ПРЕОБРАЗОВАНИЕ ТИПОВ

- **(coerce object result-type)**
 - Пытается преобразовать объект к указанному типу, если это осмысленно
 - Пример: **(coerce 3.8 'integer) -> 3**
- **(parse-integer string)**
 - Функция для преобразования строки в целое число
 - Пример: **(parse-integer "52") -> 52**
- **(format nil "~A" object)**
 - Универсальный способ преобразовать что угодно в строку
 - Пример: **(format nil "~A" 52) -> "52"**
- **Другие функции:**
 - **(float 5) -> 5.0**
 - **(character "A") -> #\A**

СРАВНЕНИЕ ЗНАЧЕНИЙ И ТИПОВ

- **(eq x y)**
 - Проверяет, являются ли **x** и **y** одним и тем же объектом в памяти
- **(eq? x y)**
 - Проверяет, являются ли либо **x** и **y** одинаковыми числами/символами, либо одним и тем же объектом
 - Пример: **(eq? 5 5) -> t**
- **(equal x y)**
 - Сравнивает структуры объектов
 - Рекурсивно устанавливает поэлементное равенство для списков и строк
 - Пример: **(equal (list 1 2) (list 1 2)) -> t**
- **(equalp x y)**
 - Более мягкая версия **equal**
 - Игнорирует регистр символов в строках и разницу между типами чисел
 - Пример: **(equalp 5 5.0) -> t**



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