

1) ООП в Scheme

В Scheme нет механизма определения классов и объектов. Но этого функционала можно добиться путем объявления функций, которые будут хранить в своем кадре вызова поля объекта и возвращать анонимную функцию, через которую можно будет обращаться к данным объекта.

Пример класса в Scheme:

```
(define (cube size)
  (lambda (var)
    (cond (leq? var 'get-volume) (* size size size))
  )
)
```

Вызов `((cube 2) 'get-volume)` вернет 8, как объем куба со стороной 2.

Но в Racket есть средства для явного создания классов:

```
(require racket/class)
(define lcg%
  (class object%
    (super-new)
    (init-field x a b m)
    (define/public (next))
    (define/public (init args))
  )
)
```

ООП подход позволяет удобно описывать различные системы, что дает возможность строить большие сложные приложения. Это достигается путем определения объектов и связей между ними.