



CHICKENscheme
a practical and portable scheme system

sECuRE, 2013-03-21

Was ist CHICKEN?

- CHICKEN is a compiler for the Scheme programming language
- Scheme ist einer der beiden großen Lisp-Dialekte
- unterstützt R5RS

Typen (primitives)

Number	42, 2.3
Boolean	#t, #f
Character	#\b
String	"foo"
Symbol	foo

Listen

(1 2 3)

(+ 1 1)

'(1 2 3)

Listen (evaluiert)

```
(1 2 3)  
; Error: call of non-procedure: 1
```

```
(+ 1 1)  
; 2
```

```
'(1 2 3)  
; (1 2 3)
```

Demo: Die REPL

Lambdas (anonyme Funktionen)

Syntax: `lambda (Liste von Parametern) Ausdrücke`

Lambdas (anonyme Funktionen)

Syntax: lambda (Liste von Parametern) Ausdrücke

```
(lambda () #f)  
; #<procedure (?)>
```

Lambdas (anonyme Funktionen)

Syntax: lambda (Liste von Parametern) Ausdrücke

```
(lambda () #f)  
; #<procedure (?)>
```

```
((lambda () #f))  
; #f
```

Lambdas (anonyme Funktionen)

Syntax: lambda (Liste von Parametern) Ausdrücke

```
(lambda () #f)  
; #<procedure (?)>
```

```
((lambda () #f))  
; #f
```

```
((lambda (x) (+ x 1)) 3)  
; 4
```

Funktionen

Syntax: define symbol value

```
(define foobar #f)  
; #<unspecified>
```

Funktionen

Syntax: define symbol value

```
(define foobar #f)  
; #<unspecified>
```

```
(define add-one  
  (lambda (x) (+ x 1)))  
; #<unspecified>
```

```
(add-one 4)  
; 5
```

Funktionen

Syntax: define symbol value

```
(define foobar #f)  
; #<unspecified>
```

```
(define add-one  
  (lambda (x) (+ x 1)))  
; #<unspecified>
```

```
(add-one 4)  
; 5
```

```
(define (add-one x)  
  (+ x 1))
```

Funktionen: Konventionen

- Prädikate enden auf ?

```
(string=? "foo" "bar")  
; #f
```

```
(symbol? 'a)  
; #t
```

- Typumwandlung mit ->

```
(json->string #f)  
; "false"
```

- „Destruktive“ Funktionen mit !

```
(define x 1)  
(set! x 2)
```

Wir bauen uns ein Programm

- Theorie lernen ist langweilig und doof
- ganze Programme schreiben ist zu schwer
- wir bauen ein einfaches Beispiel... interaktiv! :-)

Programm: $C^{1/4}$ -Planer

- Das Programm weiß über die $C^{1/4}$ im März bescheid
- Das Programm weiß, welcher Tag heute ist
- Wir implementieren, dass angezeigt wird, welche $C^{1/4}$ schon war und welche noch kommt

Betriebssystem-wechsel

Was wurde nicht behandelt?

- Andere Datentypen: alists, queues, hashes
- Input/Output, Netzwerk, etc.
- Eggs

URLs

- `http://call-cc.org/`
- `http://api.call-cc.org/doc/data-structures`
- `http://en.wikipedia.org/wiki/Scheme_(programming_language)`
- #chicken auf FreeNode