Clojure and Clojurescript

What is Clojure(script)?

→ Functional Programming Language

→ Designed for concurrency

→ Modern Lisp

→ Runs on the JVM

→ Runs on the CLR

[1] What this presentation is mostly about!

Not to be confused with:

→ Closure - CS/Math concept

→ Google Closure Compiler Project

→ Clozure CL *(another Lisp implementation)*

Rich Hickey

Syntax Examples

Data Types

```
;; Same as null in Java
nil

;; Numbers
1 1/2 2.7 1N 1M 0.666 ;; N = BigInt M = BigDecimal

;; Strings
"Dev Days 11!"
```

Operators

```
(= 1 1)
; => true

(not= 1 1)
; => false

(< 1 2)
; => true

(>= 3 4)
; => false

;; boolean or
(or (= 0 1) (= "yes" "no"))
; => false

;; boolean and
(and :free_wifi :hot_coffee)
; => :hot_coffee
```

Functions

```clojure
;; Function call
(println "Hello world!")

;; Function definition
(defn square [x]
  (* x x))

;; Anonymous functions
(map (fn [x] (* x x)) (range 1 10))
; => (1 4 9 16 25 36 49 64 81)
```

Flow control

```
(if true
  "By Thor's hammer!"
  "By Aquaman's trident!")
; => "By Thor's hammer!"

(if false
  "By Thor's hammer!"
  "By Aquaman's trident!")
; => "By Aquaman's trident!"

(when true
  (println "Success!")
  "abra cadabra")
; => Success!
; => "abra cadabra"
```

Data Structures

```
;; Maps
{:first-name "Charlie"
 :last-name "McFishwich"}

;; Vectors - O(1) lookup
[1 2 3]

;; Lists (slower performance)
'(1 2 3 4)
```

Binding a name

```
(def failed-character-names
  ["Larry Potter" "Doreen the Explorer" "The Incredible Bulk"])

;; Constant
(defonce failed-character-names
  ["Larry Potter" "Doreen the Explorer" "The Incredible Bulk"])
(defonce failed-character-names
  ["Harry Potter" "Dora the Explorer" "The Incredible Hulk"])
; failed-character-names
; => ["Larry Potter" "Doreen the Explorer" "The Incredible Bulk"]
```

Atom

→ Manage shared state

→ Synchronous

→ Used to hold immutable data

→ Free from race conditions

→ Can be updated using the `swap!` function

Reagent

→ Clojurescript library that wraps Facebook's react.js

→ Uses Clojure data structures to define HTML rather than JSX

→ Allows you to define components as functions

→ Takes care of calls to `shouldComponentUpdate`

Re-frame

→ Single `atom` used stores all state

→ Register `subscriptions` to query data

→ Write Reagent component functions to display data

→ Write and register `event handler` functions to control and update state

~~Live~~ Coding

More Info

→ Clojurescript for skeptics

→ Clojure for the Brave and True

→ Re-frame

→ Developing ClojureScript With Figwheel