

TUTORIAL CLOJURE

Primeros pasos

Leiningen

Hoy en día, la mayoría de usuarios de Clojure utilizan Leiningen para construir y gestionar sus proyectos.

Descargar: <http://leiningen.org/>

Al instalar Leiningen, se descarga automáticamente el compilador Clojure, clojure.jar

Crear un nuevo proyecto Clojure

Para crear un proyecto en Clojure con Leiningen, es necesario escribir la siguiente línea de comando:

```
lein new app clojure-noob
```

Este comando creará una carpeta llamada clojure-noob con una estructura de directorios que tiene un aspecto similar a este:

```
| .gitignore
| doc
| | intro.md
| ① | project.clj
| README.md
| ② | resources
| src
| | clojure_noob
| ③ | | | core.clj
| ④ | | test
| | | clojure_noob
| | | | core_test.clj
```

Esta estructura no es heredada de Clojure, solo es una convención usada por Leiningen. El primer archivo marcado es project.clj en ①, el cual es un archivo de configuración para Leiningen y lo ayuda a responder preguntas como "¿Qué dependencias tiene este proyecto?" y "¿Qué función se debe ejecutar en primer lugar?", en general el código fuente se salvará en

src/<project_name>. En este caso, el archivo src/clojure_noob/core.clj en ❸ es donde tendremos nuestro código. En resources ❷ es donde se guardan assets como imágenes.

Ejecutar el proyecto

En el archivo src/clojure_noob/core.clj se encuentra un println:

```
(ns clojure-noob.core
  (:gen-class))

(defn -main
  "I don't do a whole lot ... yet."
  [& args]
  (println "Hello, World!"))
```

Ejecutando lein run en la carpeta clojure_noob obtenemos nuestro Hola mundo:

```
C:\Users\jccaleroe\Downloads\Lenguajes\Clojure\clojure-noob>lein run
Hello, World!
```

Exportar el proyecto

Si se desea compartir el trabajo con personas que no tienen Leiningen se puede crear un archivo independiente que cualquiera con java puede usar, para esto ejecutar:

```
lein uberjar
```

Este comando crea el archivo target/uberjar/clojure-noob-0.1.0-SNAPSHOT-standalone.jar. el cual se puede ejecutar de esta manera:

```
java -jar target/uberjar/clojure-noob-0.1.0-SNAPSHOT-standalone.jar
```

Usando REPL

REPL es una herramienta para experimentar con código, lo hace mediante la presentación de un prompt donde se puede introducir código, lee su entrada, lo evalúa, imprime el resultado, y se repite el ciclo en donde se presenta un prompt de nuevo.

Para iniciar el servidor REPL escribimos:

```
lein repl
```

Con lo que obtenemos

```
nREPL server started on port 28925
REPL-y 0.1.10
Clojure 1.7.0

clojure-noob.core=>
```

El prompt indica que el código se ha cargado en REPL, y se pueden ejecutar las funciones que estén definidas. En este momento sólo hay una función, main la cual se puede ejecutar cambiando el println por "I'm a little teapot!":

```
clojure-noob.core=> (-main)
I'm a little teapot!
nil
```

También se pueden ejecutar funciones de Clojure, por ejemplo:

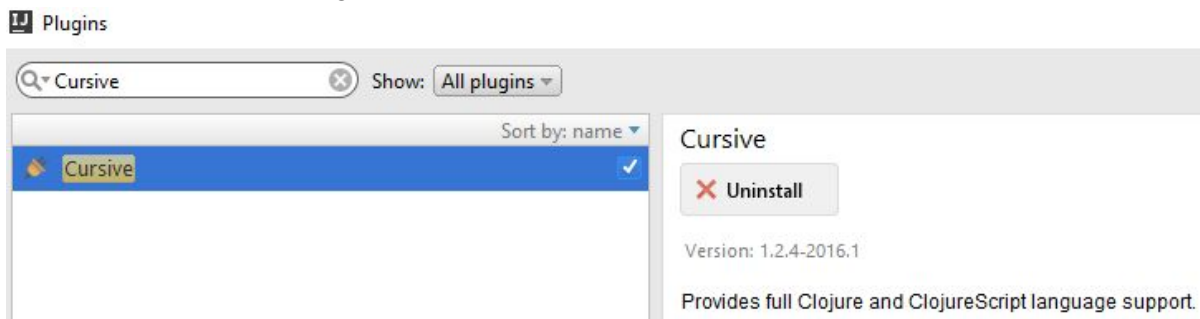
```
clojure-noob.core=> (+ 1 2 3 4)
10
clojure-noob.core=> (* 1 2 3 4)
24
clojure-noob.core=> (first [1 2 3 4])
1
```

Editores de Clojure

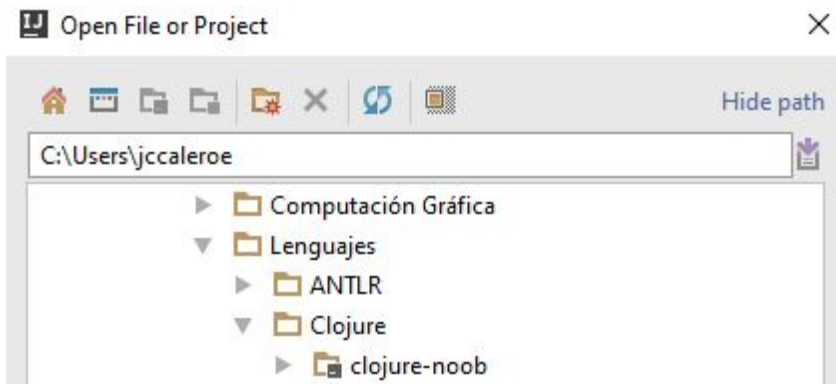
Cursive plug-in IntelliJ

Entraremos en los detalles de como correr Leiningen desde IntelliJ.

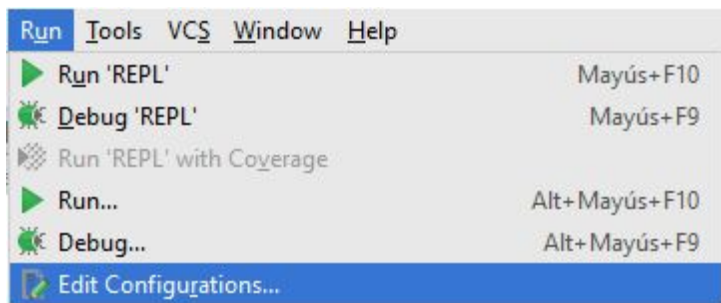
Primero instalamos el Plugin:



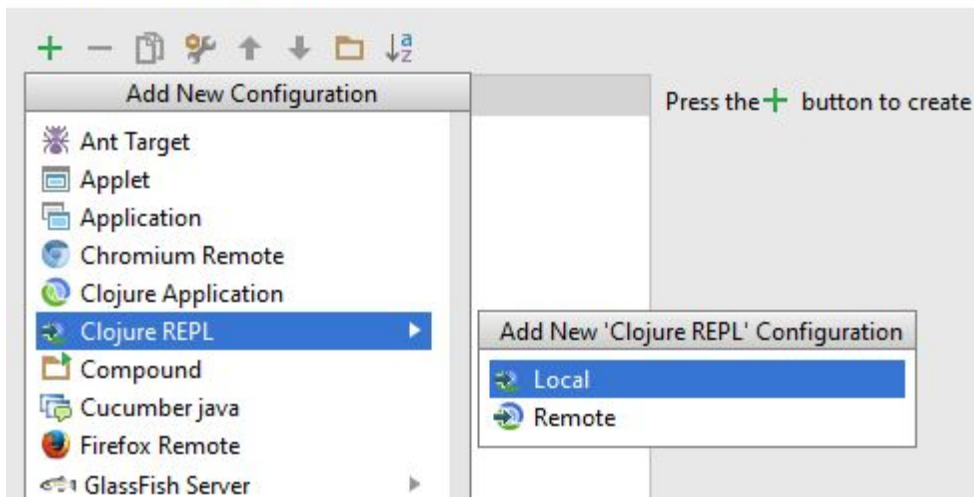
Después abrimos la carpeta clojure-noob que creamos con Leiningen e IntelliJ lo reconoce como un proyecto Leiningen:



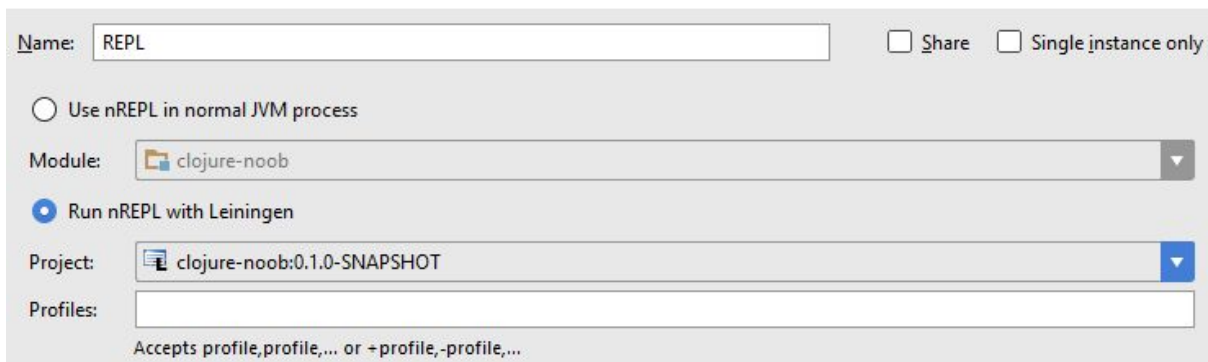
Después editamos la forma de correr el proyecto mediante REPL:



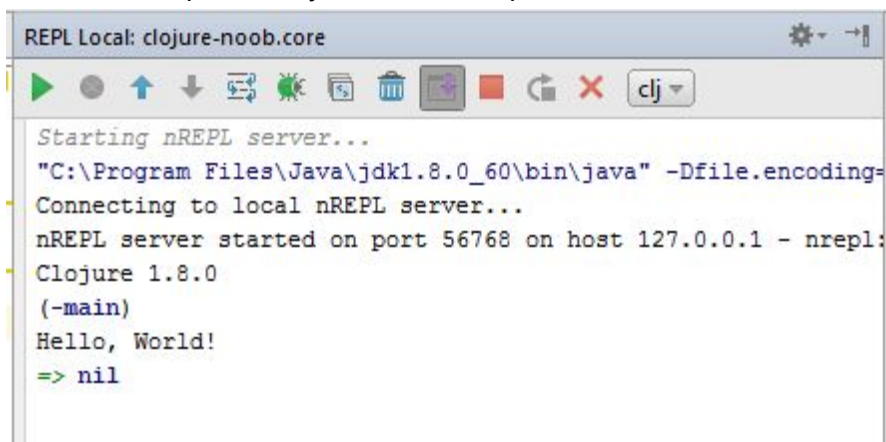
Run/Debug Configurations



Configuramos el proyecto con el modulo clojure-noob:



Corremos la aplicación y desde IntelliJ podemos usar REPL:



Otros IDEs

Este vídeo en YouTube muestra cómo configurar Sublime Text 2 para el desarrollo Clojure:

<http://www.youtube.com/watch?v=wBl0rYXQdGg/>.

Vim tiene buenas herramientas para el desarrollo Clojure. Este artículo es un buen punto de partida:

<http://mybuddymichael.com/writings/writing-clojure-with-vim-in-2013.html>.

Counterclockwise es un recomendable plugin de Eclipse:

<https://github.com/laurentpetit/ccw/wiki/GoogleCodeHome>.

Nightcode es un simple y gratuito IDE escrito en Clojure:

<https://github.com/oakes/Nightcode/>.

REPL usando clojure.jar

Descargar archivo comprimido de https://clojure.org/guides/getting_started

Descomprimir el archivo, y dentro de este correr

```
java -cp clojure-1.8.0.jar clojure.main
```

Desarrollo dinámico con una consola de evaluación REPL.

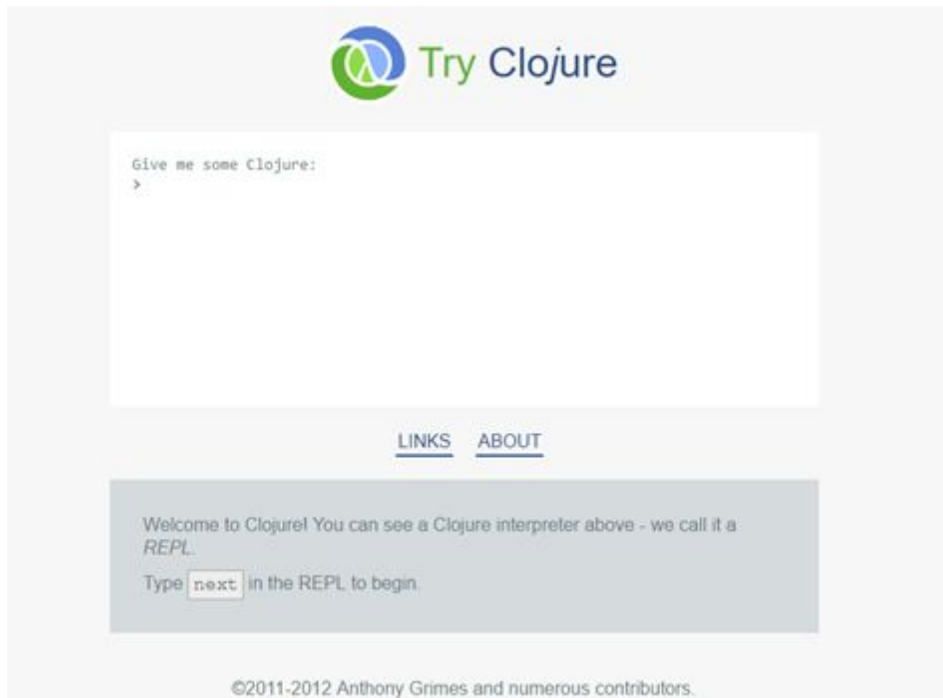
```
user=> (+ 1 2 3)
6
user=> (javax.swing.JOptionPane/showMessageDialog nil "Hello World")
```



Clojure online.

a) Try Clojure

<http://www.tryclj.com/>



b) Himera

<http://himera.herokuapp.com/index.html>



Particularidades de clojure

<https://es.wikipedia.org/wiki/Clojure>

Material sobre clojure:

<https://clojure.org/community/resources>

Tour por clojure y ejemplos

<http://www.braveclojure.com/do-things/>

<http://www.braveclojure.com/functional-programming/>