
Proyecto final

Lenguajes de programación

— Traductor de Clojure a Python —

Por: Raúl Ramírez Penagos
Jhon Torres Toloza

Introducción

Se busca realizar un traductor de Clojure a python, la idea es que se pueda utilizar como una herramienta de aprendizaje para el paradigma de programación funcional y más específicamente del lenguaje de programación Clojure.

Una solución para ello es usar la herramienta ANTLR4 para la realización del programa traductor, y generar un código limpio y adecuado.

Clotophy app

introducción:

Cuando una persona se ve en la necesidad de aprender un lenguaje de programación funcional, en este caso específico, Clojure, por lo general las nociones y lógicas de programación que maneja son la programación orientada a objetos y programación estructurada, y sabe algo de lenguajes como C++, java y python, y pasar de programar rutinas en estos lenguajes a un lenguaje diseñado principalmente para el paradigma funcional, no es tan trivial pues la sintaxis del lenguaje es muy distinta.

Clotophy app

```
1 def factor ( n):
2     """Se hace el calculo iterativamente"""
3     answer = 1
4     i = 1
5     while(i < (n + 1)):
6         answer = (i * answer)
7         i = (i + 1)
8     return answer
9 print "FACTORIAL DE LOS 10 PRIMEROS NUMEROS\n"
10 n = 1
11 while(n < 11):
12     print n , "! =" , factor(n)
13     n = (1 + n)
14
```

```
1 ▾ ( defn factor
2     "Se hace el calculo iterativamente" [n]
3     ( def answer 1 ) ( def i 1 ) ( while (< i (+ n 1)) ( def answer (* i answer) ) ( def i (+ i 1) ) ) answer)
4
5 ( println "FACTORIAL DE LOS 10 PRIMEROS NUMEROS\n" )
6 ( def n 1 )
7 ( while (< n 11) ( println n "! =" (factor n) ) ( def n (+ 1 n) ) )
```

Clotophy app

Introducción:

Por ello creemos conveniente tener una herramienta para realizar una traducción de un programa escrito en clojure, a un lenguaje que sea más “familiar” y que sea más comprensible cuando lo leamos, una muy buena alternativa para esto es python, ya que su sintaxis hace que el programa sea fácilmente entendible y eso ayudaría a que la persona en cuestión comprendiera de manera más sencilla como realizar o cómo funciona un algoritmo en clojure.

Clotophy app

Se busca desarrollar dicha herramienta haciendo uso de antlr4 como ayuda para facilitar el proceso de analisis lexico y sintactico, y enforcarnos la traduccion de las sintaxis de Clojure a Python.

Aplicaciones similares

En las búsquedas que se realizaron no se encontró como tal una implementación similar, sin embargo si existen diversas paginas web donde se menciona la importancia y la relación de ambos lenguajes y cómo realizar una traducción de uno al otro, pero está básicamente desde la parte lógica, es decir, una traducción manual.

Translating Code from Python and Scheme to Clojure

When coming to Clojure from another language, it takes a while before you start "thinking in Clojure". While ramping up, it helps to understand how to solve a problem in a language you're already familiar with, and then translate the code in some methodical fashion into Clojure.

This article will look at a simple function, remove-first, and look at how you would implement that function in Python and Scheme, and then how to methodically transform those implementations into Clojure. In all of these implementations, I'm going to ignore ways to write the function using shortcuts provided by the standard library, and focus on the implementations using standard iteration and/or recursive techniques. This will provide the clearest example of how the translation process works and can generalize to other types of functions.

Aplicación propuesta

Se va a desarrollar una aplicación la cual se encargará de traducir código en clojure a python, esto lo hará recibiendo como argumento el código fuente clojure y generando su respectivo archivo “.py”.

Se quiere realizar un análisis del lenguaje clojure, para cubrir la mayor compatibilidad y tener una buena traducción, como la definición de variables, funciones, arreglos, vectores, estructuras de datos , etc...

Implementación

Para realizar el análisis léxico y sintáctico se usó ANTLR para escribir la gramática del lenguaje y generar dichos análisis, de esta manera una vez hecha la gramática del lenguaje, resultaba más cómodo realizar el recorrido del árbol sintáctico y realizar la traducción del código pues antlr nos lo facilita.

Luego usando los métodos generados en código java, por cada regla de la gramática se realiza un recorrido y la traducción dependiendo el código fuente de Clojure pasado a la aplicación.

Implementación

```
65  @Override
66  public String visitDeclaracion(ClojureParser.DeclaracionContext ctx)
67  {
68      makeTabs( tabs );
69      file.write( ctx.ID().getText() + " = " );
70      visitExpresion( ctx.expresion() );
71      file.write( "\n" );
72      return "";
73  }
74
```

Pruebas y validación

Se crearon varios casos de entrada probando distintas declaraciones y uso de herramientas del lenguaje (Clojure) y luego de la ejecución de la traducción se procede a ejecutar el código python resultante y se verifica que las salidas del programa sean las esperadas y que este no contenga errores de ejecución o interpretación.

Conclusiones

Contar con una herramienta de traducción es muy útil para el proceso de aprendizaje ya que nos ahorra tiempo y nos ayuda comprender de forma más rápida un código de un lenguaje en cuestión.

Clojure al comienzo puede ser algo confuso ya que no solo es una sintaxis nueva la que se debe aprender, también es pensar en programación en Clojure, por ejemplo, la sentencia “for” en este contexto es una operación sobre listas, pero en otros lenguajes es un ciclo que se puede usar para múltiples propósitos, estas son algunas cosas que se deben tener en cuenta y que no se obvian de una traducción.

Referencias

<http://clojure.org/reference/reader>

<http://programming-puzzler.blogspot.com.co/2010/07/translating-code-from-python-and-scheme.html>

<https://jkkramer.wordpress.com/2011/03/29/clojure-python-side-by-side/>