

# PyCharm ANTLR4 Guide

Karl Janson

November 1, 2019

## 1 Introduction

In order to work with ANTLR4 from an IDE, we first need to install the ANTLR4 grammar plugin. In our case, we are going to write our code in Python3 and use the PyCharm IDE (based on IntelliJ). However, ANTLR4 has plugins available for most mainstream IDEs and ANTLR supports most of the modern languages.

*Note:* The plugin is going installed as the local user. This means that when working in the class room, you might have to re-install install the plugin every time you log on.

## 2 Environment Setup

### 1. Download the ANTLR v4 grammar plugin for PyCharm

- (a) Open the following link in your browser: [\[LINK\]](#)
- (b) Scroll down and download the version fitting your IDE (note down the path where the file gets downloaded)

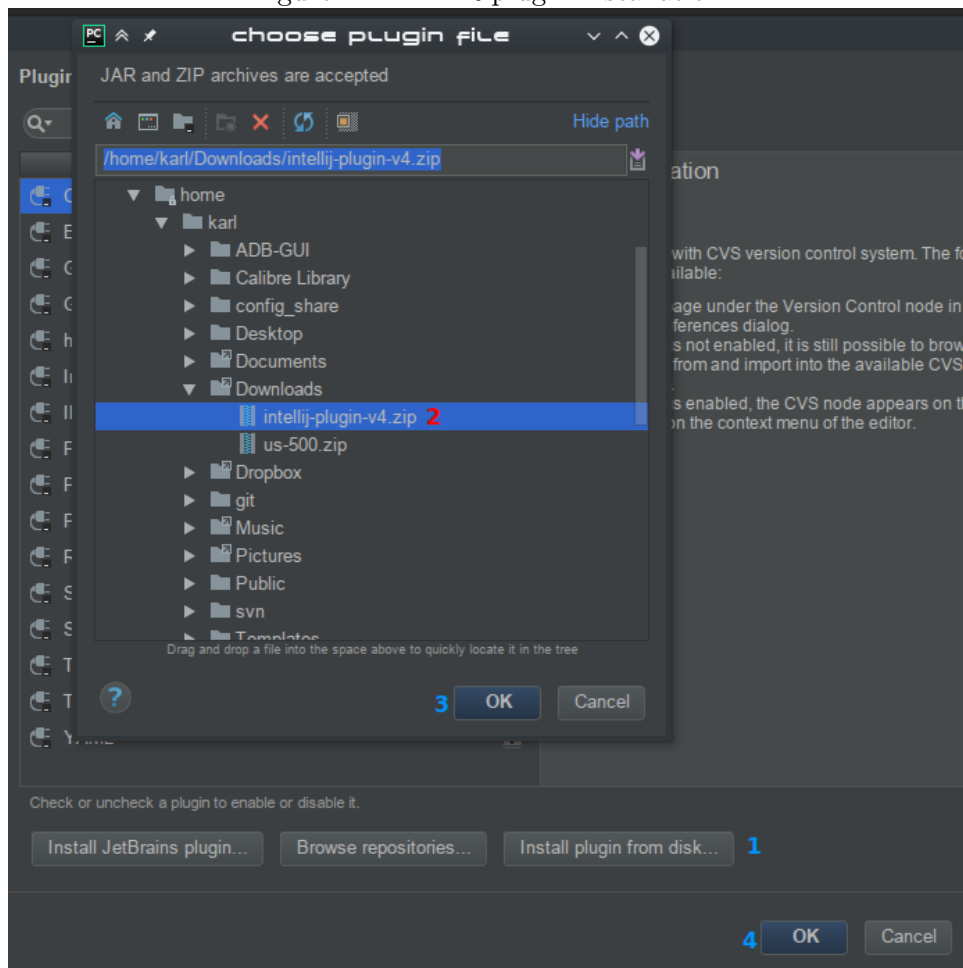
### 2. Create a new project in PyCharm

- (a) Call the project **ANTLR\_CSV\_to\_JSON**
- (b) Select **Python3** as the interpreter

### 3. Install the plugin

- (a) Open PyCharm settings window (press CTRL+ALT+S)
- (b) Open **Plugins** page
- (c) Click **Install plugin from disk. . .**
- (d) Choose the zip file containing the plugin that you just downloaded (Figure 1)
- (e) Click OK
- (f) Restart PyCharm to activate the plugin

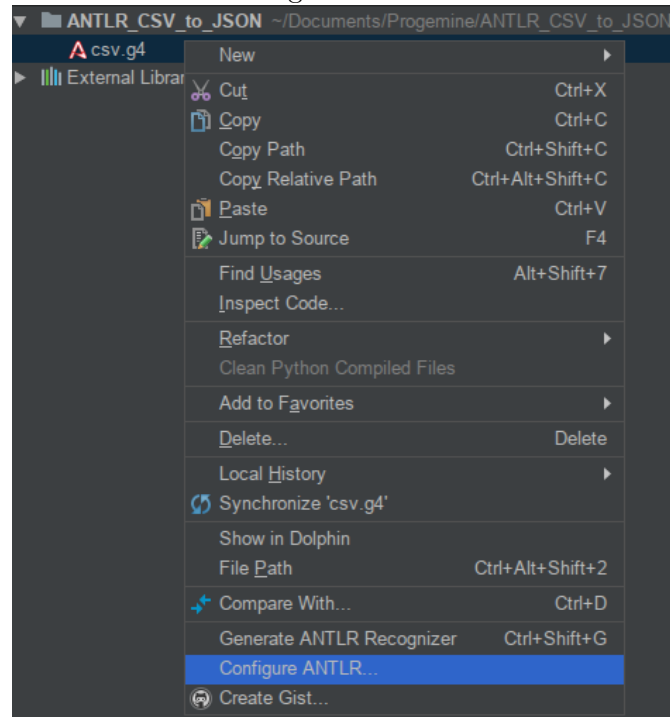
Figure 1: ANTLR plugin installation



### 3 Creating a New ANTLR File

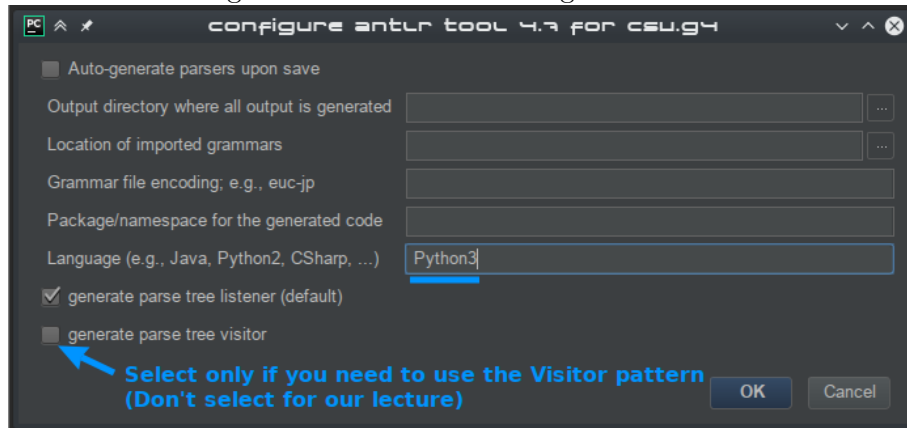
1. Right-click on your project folder in PyCharm, select **New** → **File**
2. Enter `<grammar_name>.g4` as the file name (Ex: `csv.g4`)
3. If PyCharm offers to install syntax highlighting support for `.g4` files, install it and restart PyCharm
4. Configure ANTLR for the current grammar (Figure 2):
  - (a) Right click on the grammar file and choose **Configure ANTLR. . .**

Figure 2: Selection of “Configure ANTLR” from the context menu



(b) In the opened dialog specify **Python3** as language (Figure 3).

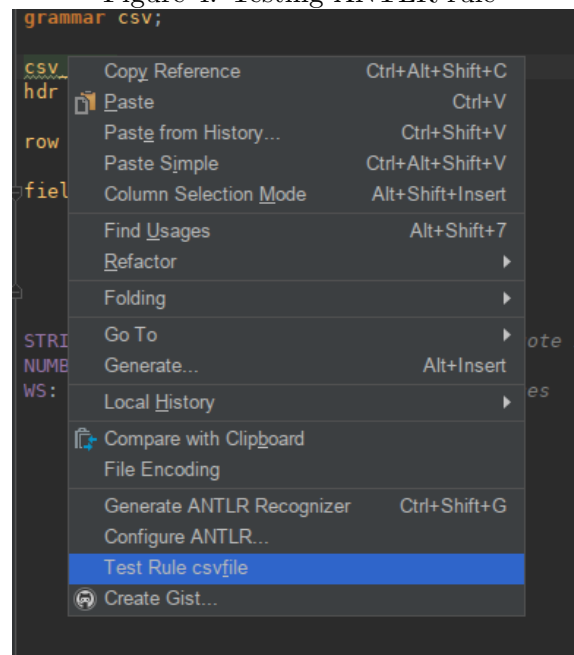
Figure 3: ANTLR tool configuration window



## 4 Testing ANTLR Rules

Right click on the top-level parser rule and choose  
**Test Rule <rule\_name>** (Figure 4)

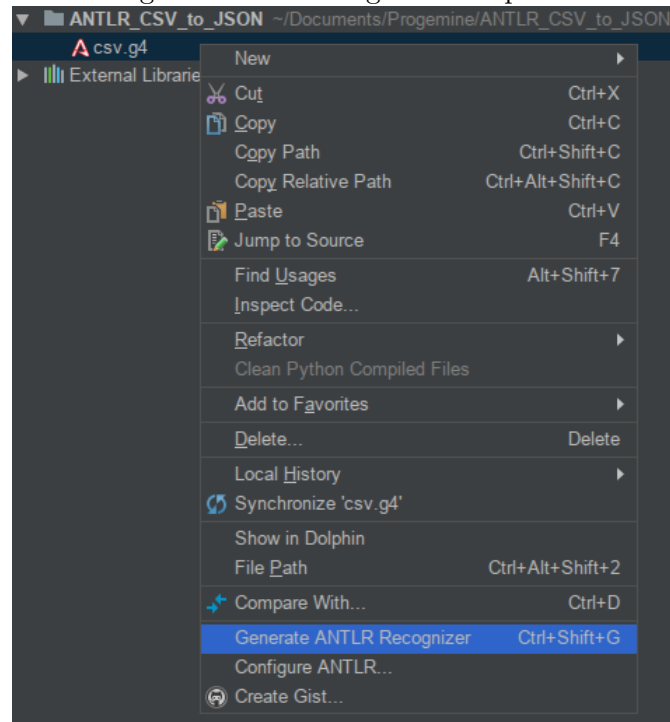
Figure 4: Testing ANTLR rule



## 5 Generating an ANTLR4 Parser

Right click on your **.g4** file and choose **Generate ANTLR Recognizer** from the menu (Figure 5).

Figure 5: Generating ANTLR parser



### Debugging information

If you get the following error:

**symbol file conflicts with generated code in target language or runtime**

Then it means that you have specified a parser rule in your **.g4** file with a name that shadows a Python's built-in function name and thus it would be redefined. Please rename your rule.