# PseuToPy: Towards a Non-English Natural Programming Language

Patrick Wang
Institut Supérieur d'Électronique de Paris
Human-Machine Interaction Research Group

✉ patrick.wang@isep.fr
https://pseutopy.com
@Gkz_

## INTRODUCTION

When learning to program, one must master the grammar and semantics of a programming language. This task can be difficult for multiple reasons. First, the syntax and grammar rules of a programming language yields to instructions that are unnatural. Second, popular text-based programming languages taught in CS0/1(e.g., Python, Java, C/C++, Javascript) use English words in their grammar specifications. According to Qian & Lehman [1], this represents a real obstacle as English proficiency seems to be a good predictor of success in introductory programming courses. Knowing that less than 20% of the world population speak English as a first or second language (see Figure 1) [2], what can we do to help non-English speakers learn to program?

| Language | L1 speakers | L2 speakers | Total |
|---|---|---|---|
| English | 369.9 million | 978.2 million | 1.348 billion |
| Mandarin Chinese | 921.2 million | 198.7 million | 1.120 billion |
| Hindi | 342.2 million | 258.3 million | 600 million |
| Spanish | 471.4 million | 71.5 million | 543 million |
| Standard Arabic | N/A | 274 million | 274 million |

Table 1: Top five languages spoken according to Ethnologue (2021, 24th edition).

**Research Question:** Would programming beginners learn better to program with a programming language that uses their native tongue and generates natural programming instructions?

## WHAT IF ENGLISH WASN'T USED?

Imagine that instead of using English words, programming languages were to use Chinese. Would you be able to infer the output of this program? To write this program without making syntax errors? If you don't know Chinese, then probably not.

```
设 guojiamingdan 赋值 ["法国", "中国", "美国", "西班牙",  "英国"]
所有 guojia 在 guojiamingdan 中执行:
    打印(guojia)
```

Congratulations, you are now in the shoes of all non-English speakers who wish to learn Python!

## GRAMMAR SPECIFICATION OF PSEUTOPY

The design of PseuToPy relies on five aspects:

1. The formal grammar of PseuToPy is based on the grammar specification for Python;
2. This formal grammar is modified to add alternative instructions in another language;
3. The alternative instructions can help disambiguate symbols or emphasize on programming concepts;
4. The instructions produced resemble almost grammatically correct natural language sentences;
5. Instructions written in PseuToPy can be converted into Python code and then ran to show learners the result of its execution.

Below is the "assign" rule that describes how Python handles assign statements:

```
?assign: (testlist_star_expr
                ("=" (yield_expr|testlist_star_expr))*)
```

This rule can be modified to add an alternative French version, which also helps disambiguate the equal symbol. Note that the original Python version is still considered valid with this new rule.

```
?assign: (testlist_star_expr
                ("=" (yield_expr|testlist_star_expr))*)
         | ("mettre" testlist_star_expr
                ("à" (yield_expr | testlist_star_expr))*)
```

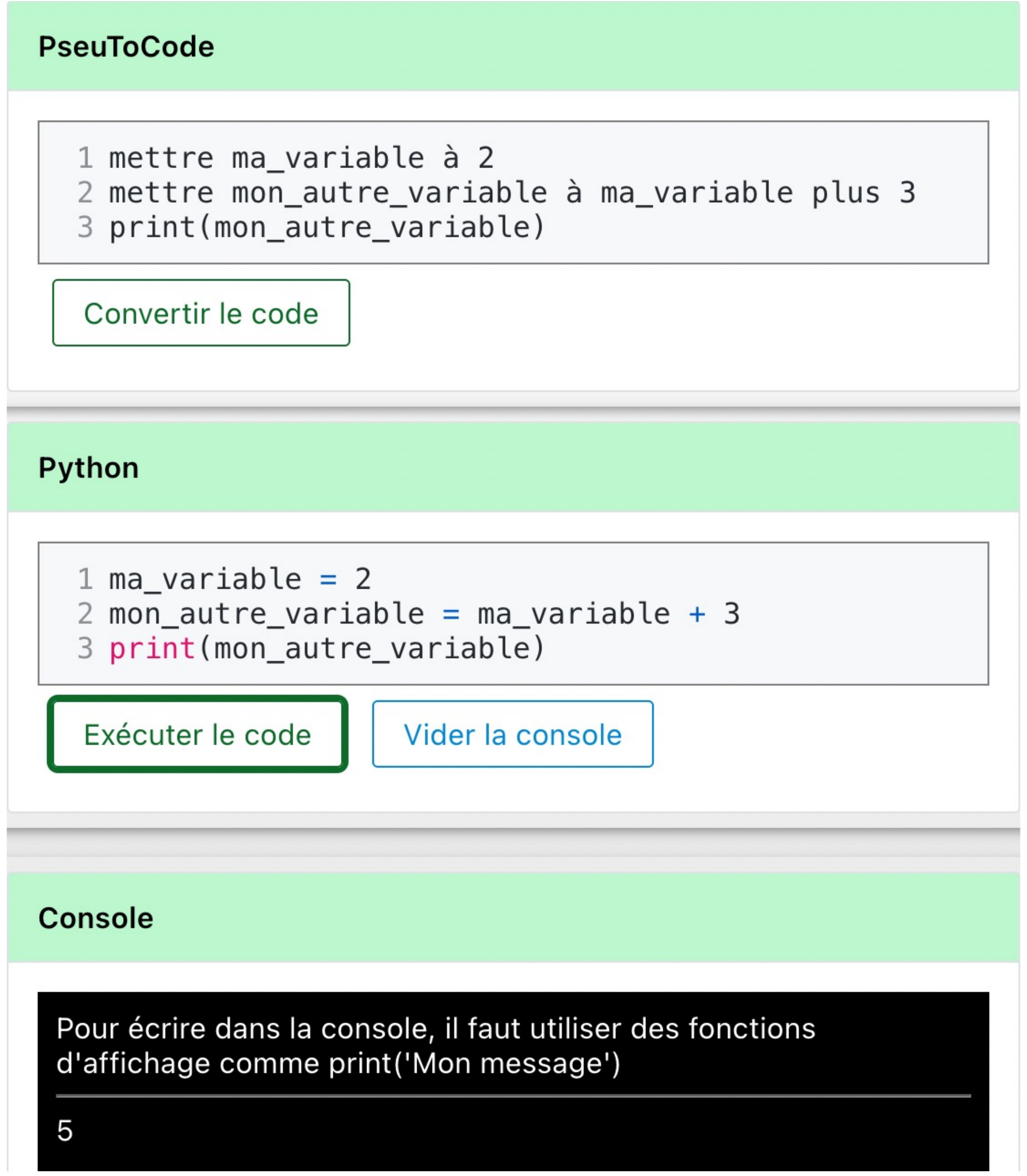
This rule can be modified to add an alternative French version, which also helps disambiguate the equal symbol. Note that the original Python version is still considered valid with this new rule. The same work can be done with other languages such as Chinese to demonstrate the possibility to work with non UTF-8 characters.

```
?assign: (testlist_star_expr
                ("=" (yield_expr|testlist_star_expr))*)
         | ("设" testlist_star_expr
                ("赋值" (yield_expr | testlist_star_expr))*)
```

With these rules, the following instructions are all equivalent:

```
my_variable = 42
mettre my_variable à 42
设 my_variable 赋值 42
```

## IMPLEMENTATION OF PSEUTOPY



PseuToCode

```
mettre ma_variable à 2
mettre mon_autre_variable à ma_variable plus 3
print(mon_autre_variable)
```

Python

```
ma_variable = 2
mon_autre_variable = ma_variable + 3
print(mon_autre_variable)
```

Console

```
Pour écrire dans la console, il faut utiliser des fonctions
d'affichage comme print('Mon message')

5
```

## CONCLUSION AND FUTURE WORKS

With PseuToPy, we wish to offer non-English speakers a tool to help them learn to program. Currently, only a French version exists, and we are looking for support from the community to translate PseuToPy in other languages. We also want to create learning resources that could be used in middle and high schools in France.

ICER 2021

**REFERENCES**

[1] Qian, Y., & Lehman, J. (2016). Correlates of Success in Introductory Programming: A Study with Middle School Students. Journal of Education and Learning 5, 2 (2016), 73–83.

[2] https://en.wikipedia.org/wiki/List_of_languages_by_total_number_of_speakers

isep
École d'ingénieurs du numérique