

# Good Coding Practices for Pytnam Project Team

## 0. Tools

- We use Python 3.x (3.5.2 when writing this document), the Anaconda distribution: <https://www.continuum.io/anaconda-overview>.
- Github repository for Pytnam: <https://github.com/pytnam/pytnam>
- The default operating system is Windows. The code, however, should be kept as platform-agnostic as possible.
- The recommended IDE is JetBrains PyCharm 2017. Free academic license can be obtained at <https://www.jetbrains.com/student>.

## 1. Indentation & Limits

- Code indentation should be done ONLY with spaces.
- 4 spaces should be used per indentation level. The four-space rule is optional in line continuation, but this possibility must not be abused.
- The maximum line length is 79 characters.
- If breaking is necessary, line should always break BEFORE the binary operator and not before the second operand.

## 2. Blank Lines

- Class definitions and top-level functions should be surrounded with two blank lines.
- Method definitions inside a class should be surrounded with one blank line.
- Inside the method body, logical sections of code should be separated by one blank line.
- Blank lines may be omitted between logically related one-line functions.

## 3. Encoding

- The only legitimate file encoding is UTF-8. Pure Practical Reason & Human Dignity does not support Windows-1252.

## 4. Imports

- All import statements should be included at the beginning of the file. PyCharm's default import order is fine.

## 5. Whitespace Use

- Outside mathematical expressions, brackets should never be preceded by a whitespace.
- The comma operator should always be followed by a whitespace.
- The following operators should always be surrounded with single whitespaces: {=, +=, ==, ->, <, >, !=, <>, <=, >=, in, not in, is, is not, not, and, or}. The only exception is = operator setting default parameters in functions.
- Mathematical expressions should be written reasonably. In general, too many whitespaces is always better than too few.

## 6. Naming Conventions

- Package names should be short and all lowercase, e.g. pckgname.
- Class names should obey CapWords convention, e.g. ThisIsAClassName.
- Exception names should obey CapWords convention, e.g. BadException.
- Function names should be lowercase with underscores (long function names are discouraged due to readability), e.g. my\_fancy\_function(). Note that the 'self' keyword should always come as the first argument to method.
- Variable and argument names should be lowercase with underscores, e.g. numeric\_arg. Descriptive, readable names are strongly encouraged.
- Constants should be all capitalized with underscores, e.g. PLANCK\_CONSTANT.

## 7. Other Issues

- In cases not mentioned in this guide (which are plenty), please refer to Pure Practical Reason & Human Dignity as well as to PEP8 (the official Python style guidelines). In case of rule conflict, the former has ALWAYS priority over the latter.