## Tools for refactoring python to c++:

1. Cython: Cython is a superset of Python that allows you to write C-like code within Python files. It can help with performance optimization, but it doesn't convert Python to C++ directly.
2. SWIG (Simplified Wrapper and Interface Generator): SWIG is a tool that can generate wrapper code to interface C++ code with various programming languages, including Python. It doesn't automatically refactor Python into C++, but it helps in creating a bridge between the two.
3. Pybind11: Pybind11 is a lightweight header-only library that simplifies the process of exposing C++ code to Python and vice versa. Similar to SWIG, it doesn't refactor Python code to C++, but it facilitates integration between the two languages.
4. Nuitka: Nuitka is a Python compiler that translates Python code to C or C++ code and then compiles it. While it doesn't directly produce readable C++ code, it can enhance the performance of your Python code by generating compiled binaries.
5. Hand-coding: Given the fundamental differences between Python and C++, refactoring often involves a manual process. You may need to rewrite significant portions of the code, taking into account differences in memory management, data structures, and language features.