**Python工程编译打包**

1. **安装依赖**

|  |
| --- |
| Shell apt install patchelf pip install nuitka |

2. **编译**

2.1 **将非主程序入口文件编译成 .so 文件**

|  |
| --- |
| Bash find . -name "\*.py" \  ! -name "main.py" \  ! -name "\_\_init\_\_.py" \ | xargs -P 8 -I {} sh -c '  d=$(dirname "{}")  nuitka --module "{}" --output-dir="$d" --lto=no ' |

其中 main.py 为主程序入口文件。

注：不同项目的主程序入口文件名可能不同，请根据实际情况使用对应的文件名。

2.2 **将主程序入口文件编译成 .bin 文件**

|  |
| --- |
| Bash nuitka main.py --lto=no |

命令执行结束后，会生成一个 main.bin 文件。

注：不同项目的生成的 .bin 文件名可能不同，请根据实际情况使用对应的文件名。

3. **删除编译缓存及py源文件**

编译完成后，使用以下命令清理编译生成的中间文件及项目源码（.py）文件。由于已生成对应的 .so 文件或 .bin文件，故无需再保留 .py 源码（请提前备份 .py 文件，以防需要回溯源代码）：

|  |
| --- |
| Bash find . \  \( -name "\*.build" \  -o -name "\*.pyi" \  -o -name "\*.pyc" \  -o \( -name "\*.py" ! -name "\_\_init\_\_.py" \) \  \) \  -exec rm -rf {} + |

4. **打包**

使用 shell 压缩打包（如zip、tar等）命令将当前目录打包即可。

5. **运行**

1. 解压第“4”步骤生成的压缩文件
2. 进入解压后的目录执行:

|  |
| --- |
| Bash ./main.bin |

注：不同项目的 .bin 文件名可能不同，请根据实际情况使用对应的文件名。