
fast-package-file

Release 1.1

Jul 06, 2019

Contents

1	Installation	3
2	Features	5
3	Contribute	7
4	License	9

Package a directory to a file, with fast file access and compression support

```
import fast_package_file

# Package a directory into a file
fast_package_file.build('a_directory', 'a_package.file')

# Prepare a package file
data_package = fast_package_file.PackagedDataFile('a_package.file')

# Load a file from the packed directory and save it
with open('any.file', 'wb') as any_file:
    data_package.load_file('path\\to\\any.file')

# Or just get the raw binary data
from PIL import Image
i = Image.open(io.BytesIO(data_package.load_file('image.png'))))

# Some other useful functions
data_package.load_bulk(prefix='audio\\sfx\\', postfix='.wav')
fast_package_file.oneshot('a_package.file', 'path\\to\\any.file')
fast_package_file.oneshot_bulk('a_package.file', prefix='audio\\sfx\\', postfix='.wav
→')
```


CHAPTER 1

Installation

From PyPI:

```
pip install fast-package-file
```

Or from Github:

```
pip install git+git://github.com/Kataiser/fast-package-file.git@master#egg=fast_  
↪package_file
```


CHAPTER 2

Features

- Is fast because only the data needed is loaded from the package file, total package size is irrelevant
- Obfuscates files from (most) users
- Like a .zip file, but doesn't decompress the entire thing when reading just one file
- Includes the entire directory and subdirectories, not just surface-level files
- Files are compressed with Gzip, but only if compression improves file size (per file) and is enabled (per package file)
- Pretty good error handling when loading package files, just catch `fast_package_file.PackageDataError`
- Inspired by video game packaging, such as UE4's .pak or GTA V's .rpf formats
- Cross-platform, has CI for Linux, MacOS, and Windows

CHAPTER 3

Contribute

- Issue Tracker: <https://github.com/Kataiser/fast-package-file/issues>
- Source Code: <https://github.com/Kataiser/fast-package-file>

CHAPTER 4

License

The project is licensed under the MIT license.