



Поиск проектов

[Помощь](#)[Спонсоры](#)[Войти](#)[Зарегистрироваться](#)

python-dotenv 1.0.0

Последняя
версия`pip install python-dotenv` Выпущен: 24 февр.
2023 г.

Read key-value pairs from a `.env` file and set them as environment variables

Навигация

[Описание
проекта](#) [История
выпусков](#) [Загрузка
файлов](#)

Ссылки проекта

[Homepage](#)

Статистика

Статистика GitHub:

[Звёзд: 6343](#)

Описание проекта

python-dotenv

 Run Tests **passing** **pypi package** **1.0.0**

Python-dotenv считывает пары ключ-значение из `.env` файла и может устанавливать их в качестве переменных среды. Это помогает в разработке приложений, следующих [12-факторным](#) принципам.


- [Приступая к работе](#)
- [Другие варианты использования](#)
 - [Загружайте конфигурацию без изменения среды](#)
 - [Анализировать конфигурацию в виде потока](#)
 - [Загрузите файлы `.env` в IPython](#)
- [Интерфейс командной строки](#)
- [Формат файла](#)
 - [Многострочные значения](#)
 - [Расширение переменной](#)
- [Связанные проекты](#)

 Форков: 400 Open
issues: 30 Open PRs:
13

Смотрите статистику этого проекта на [Libraries.io](https://libraries.io) или в нашем общедоступном наборе данных на [Google BigQuery](https://google.com/bigquery)

Метаданные

Лицензия: BSD
License (BSD-3-Clause)

Автор: [Saurabh Kumar](#) 

environment variables, deployments, settings, env, dotenv, configurations, python

Требуется: Python
>=3.8

Сопровождающие

[bbc](#)[theskumar](#)

Классификаторы

- [Благодарности](#)

Приступая к работе

установить с помощью `pip python-dotenv`

Если ваше приложение берет свою конфигурацию из переменных окружения, например, 12-факторное приложение, запускать его в процессе разработки не очень практично, потому что вам приходится устанавливать эти переменные окружения самостоятельно.

Чтобы помочь вам в этом, вы можете добавить Python-dotenv в свое приложение, чтобы заставить его загружать конфигурацию из `.env` файла, когда он присутствует (например, в разработке), оставаясь настраиваемым через среду:

```
from dotenv import load_dotenv

load_dotenv() # take environment variables from .env.

# Code of your application, which uses environment variables
# `os.getenv`) as if they came from the actual environment
```

By default, `load_dotenv` doesn't override existing environment variables.

Чтобы настроить среду разработки, добавьте `.env` в корневой каталог вашего проекта:

```
.
├── .env
└── foo.py
```

Синтаксис `.env` файлов, поддерживаемых python-dotenv, аналогичен синтаксису Bash:

```
# Development settings
DOMAIN=example.org
```

Статус разработки

- о 5 -
Производственный
/ стабильный

Окружающая среда

- о Веб-среда

Целевая аудитория

- о Разработчики
- о Системные администраторы

Лицензия

- о Одобрено OSI ::
Лицензия BSD

Операционная
система

- о Не зависит от
операционной
системы

Язык

программирования

- о Python
- о Python :: 3
- о Python :: 3.8
- о Python :: 3.9
- о Python :: 3.10
- о Python :: 3.11
- о Python :: 3.12
- о Python ::
Реализация ::
PyPy

Тема

- о Система ::
Системное
администрирование
- о Служебные
программы



Capital One
является

```
ADMIN_EMAIL=admin@${DOMAIN}
ROOT_URL=${DOMAIN}/app
```

Если вы используете переменные в значениях, убедитесь, что они окружены { и }, как \${DOMAIN}, поскольку пустые переменные, такие как \$DOMAIN, не расширяются.

Вы, вероятно, захотите добавить .env в свой .gitignore, особенно если он содержит секреты, такие как пароль.

Смотрите раздел "Формат файла" ниже для получения дополнительной информации о том, что вы можете записать в .env файл.

Другие варианты использования

Загружайте конфигурацию без изменения среды

Функция dotenv_values работает более или менее так же, как load_dotenv, за исключением того, что она не затрагивает среду, она просто возвращает a dict со значениями, проанализированными из .env файла.

```
from dotenv import dotenv_values

config = dotenv_values(".env") # config = {"USER": "fo
```

Это, в частности, обеспечивает расширенное управление конфигурацией:

```
import os
from dotenv import dotenv_values

config = {
    **dotenv_values(".env.shared"), # load shared develop
    **dotenv_values(".env.secret"), # load sensitive va
    **os.environ, # override loaded values with environ
}
```

Анализировать конфигурацию в виде потока

постоянным
спонсором
Python Software
Foundation.

Спонсор *PSF*
Обслуживается
этично

`load_dotenv` и `dotenv_values` принимают **потоки** через их `stream` аргумент. Таким образом, можно загружать переменные из источников, отличных от файловой системы (например, сети).

```
from io import StringIO

from dotenv import load_dotenv

config = StringIO("USER=foo\nEMAIL=foo@example.org")
load_dotenv(stream=config)
```

Загрузите файлы .env в IPython

Вы можете использовать dotenv в IPython. По умолчанию он будет использоваться `find_dotenv` для поиска `.env` файла:

```
%load_ext dotenv
%dotenv
```

Вы также можете указать путь:

```
%dotenv relative/or/absolute/path/to/.env
```

Необязательные флаги:

- `-o` для переопределения существующих переменных.
- `-v` для увеличения детализации.

Интерфейс командной строки

Также включен интерфейс CLI `dotenv`, который помогает вам управлять `.env` файлом, не открывая его вручную.

```
$ pip install "python-dotenv[cli]"
$ dotenv set USER foo
$ dotenv set EMAIL foo@example.org
$ dotenv list
USER=foo
```

```
EMAIL=foo@example.org
$ dotenv list --format=json
{
  "USER": "foo",
  "EMAIL": "foo@example.org"
}
$ dotenv run -- python foo.py
```

Запустите `dotenv --help` для получения дополнительной информации о параметрах и подкомандах.

Формат файла

Формат официально не указан и со временем все еще улучшается. При этом `.env` файлы должны в основном выглядеть как файлы Bash.

Ключи могут быть без кавычек или в одинарных кавычках. Значения могут быть без кавычек, в одинарных или двойных кавычках. Пробелы перед и после клавиш, знаки равенства и значения игнорируются. За значениями может следовать комментарий. Строки могут начинаться с `export` директивы, что не влияет на их интерпретацию.

Разрешенные escape-последовательности:

- в значениях, заключенных в одинарные кавычки: `\\`, `\'`
- в значениях, заключенных в двойные кавычки: `\\`, `\'`, `\"`, `\a`, `\b`, `\f`, `\n`, `\r`, `\t`, `\v`

Многострочные значения

Значения, заключенные в одинарные или двойные кавычки, могут занимать несколько строк. Следующие примеры эквивалентны:

```
FOO="first line
second line"
```

```
FOO="first line\nsecond line"
```

Переменная без значения

Переменная не может иметь значения:

```
FOO
```

Это приводит к `dotenv_values` связыванию имени этой переменной со значением `None` (например, `{"FOO": None}`). `load_dotenv`, с другой стороны, такие переменные просто игнорируются.

Это не следует путать с `FOO=`, в этом случае переменная связана с пустой строкой.

Расширение переменной

Python-dotenv может интерполировать переменные, используя расширение переменной POSIX.

С помощью `load_dotenv(override=True)` или `dotenv_values()` значение переменной является первым из значений, определенных в следующем списке:

- Значение этой переменной в `.env` файле.
- Значение этой переменной в среде.
- Значение по умолчанию, если оно указано.
- Пустая строка.

С `load_dotenv(override=False)` значением переменной является первое из значений, определенных в следующем списке:

- Значение этой переменной в среде.
- Значение этой переменной в `.env` файле.
- Значение по умолчанию, если оно указано.
- Пустая строка.

Связанные проекты

- [Honcho](https://github.com/django/honcho) - Для управления приложениями на основе Procfile.

- [django-dotenv](#)
- [django-окружающая среда](#)
- [django-окружающая среда-2](#)
- [django-конфигурация](#)
- [дамп-env](#)
- [окрестности](#)
- [dynaconf](#)
- [синтаксический анализ_it](#)
- [python-развязка](#)

Благодарности

В настоящее время этот проект поддерживается [Саурабхом Кумаром](#) и [Бертраном Боннефуа-Клоде](#) и был бы невозможен без поддержки этих [замечательных людей](#).

Журнал изменений

Все заметные изменения в этом проекте будут задокументированы в этом файле.

Формат основан на [ведении журнала изменений](#), и этот проект придерживается [семантического управления версиями](#).

1.0.0

Исправлено

- Прекратите поддержку python 3.7, добавьте python 3.12-dev (#449 от [@theskumar](#))
- Обращать внимание на ситуации, когда cwd не существует. (#446 от [@jctanner](#))

0.21.1 - 2022-01-21

Добавлен

- Используйте Python 3.11, не являющийся бета-версией, в CI (# 438 от [@bbc2](#))

- Модернизация кода переменных (#434 от [@Nougat-Waffle](#))
- Модернизировать main.py и parser.py код (#435 от [@Nougat-Waffle](#))
- Улучшить лаконичность cli.py и инициализировать.py (#439 от [@Nougat-Waffle](#))
- Улучшено сообщение об ошибке для `get` и `list` команд, когда файл env не может быть открыт (# 441 от [@bbc2](#))
- Обновлено лицензия для приведения в соответствие с шаблоном BSD OSI (# 433 от [@lsmith77](#))

Исправлено

- Исправлена ошибка вне области видимости, когда переменная "dest" не определена (#413 от [@theGOTOguy](#))
- Исправлено предупреждение теста IPython об устаревшем `magic` (# 440 от [@bbc2](#))
- Исправьте подсказку о типе для переменной `dotenv_path`, добавьте псевдоним `strPath` (#432 от [@eaf](#))

0.21.0 - 2022-09-03

Добавлен

- CLI: добавить поддержку вызовов через 'python -m'. (# 395 от [@theskumar](#))
- `load_dotenv` теперь функция возвращается `False`. (#388 от [@larsks](#))
- CLI: добавить параметр `--format=` в команду `list`. (#407 от [@sammck](#))

Исправлено

- Удалите Python 3.5 и 3.6 и обновите GA (# 393 от [@eggplants](#))
- Используйте `open` вместо `io.open`. (#389 от [@rabinadk1](#))
- Улучшена документация для переменных без значения (#390 от [@bbc2](#))
- Добавить `parse_it` в связанные проекты (# 410 от [@naorlivne](#))

- Обновление README.md (# 415 от [@harveer07](#))
- Улучшить документацию с прямым использованием MkDocs (#398 от [@bbc2](#))

0.20.0 - 2022-03-24

Добавлен

- Добавьте параметр `encoding` (`Optional[str]`) в `get_key`, `set_key` и `unset_key`. (#379 от [@bbc2](#))

Исправлено

- Используйте dict для указания `entry_points` параметра `setuptools.setup` (#376 от [@mgorny](#)).
- Не создавайте универсальные диски (#387 от [@bbc2](#)).

0.19.2 - 2021-11-11

Исправлено

- В `set_key` при необходимости добавьте отсутствующий символ новой строки перед новой записью. (#361 от [@bbc2](#))

0.19.1 - 2021-08-09

Добавлен

- Добавлена поддержка Python 3.10. (#359 от [@theskumar](#))

0.19.0 - 2021-07-24

Изменен

- Требуется Python 3.5 или более поздняя версия. Python 2 и 3.4 больше не поддерживаются. (#341 от [@bbc2](#)).

Добавлен

- `dotenv_path` Аргумент `set_key` и `unset_key` теперь имеет тип `Union[str, os.PathLike]` вместо просто `os.PathLike` (#347 от [@bbc2](#)).
- `stream` Аргументом `load_dotenv` и `dotenv_values` теперь может быть текстовый поток (`IO[str]`), который включает в себя такие значения, как `io.StringIO("foo")` и `open("file.env", "r")` (#348 от [@bbc2](#)).

0.18.0 - 2021-06-20

Изменен

- Поднимите, `ValueError` если `quote_mode` это не один из `always`, `auto` или `never` в `set_key` (#330 от [@bbc2](#)).
- При записи значения в файл `.env` с помощью `set_key` или `dotenv set <key> <value>` (#330 от [@bbc2](#)):
 - Используйте одинарные кавычки вместо двойных.
 - Не удаляйте окружающие кавычки.
 - В `auto` режиме не добавляйте кавычки, если значение состоит только из буквенно-цифровых символов (как определено `string.isalnum`).

0.17.1 - 2021-04-29

Исправлено

- Исправлены тесты для сред сборки, основанные на `PYTHONPATH` (# 318 от [@befeleme](#)).

0.17.0 - 2021-04-02

Изменен

- Сделайте так, чтобы `dotenv get <key>` отображалось только значение, а не `key=value` (#313 от [@bbc2](#)).

Добавлен

- Добавьте параметр `--override/--no-override` в `dotenv run` (# 312 от [@zueve](#) и [@bbc2](#)).

0.16.0 - 2021-03-27

Изменен

- The default value of the `encoding` parameter for `load_dotenv` and `dotenv_values` is now `"utf-8"` instead of `None` (#306 by [@bbc2](#)).
- Fix resolution order in variable expansion with `override=False` (#287 by [@bbc2](#)).

0.15.0 - 2020-10-28

Added

- Add `--export` option to `set` to make it prepend the binding with `export` (#270 by [@jadutter](#)).

Changed

- Make `set` command create the `.env` file in the current directory if no `.env` file was found (#270 by [@jadutter](#)).

Fixed

- Fix potentially empty expanded value for duplicate key (#260 by [@bbc2](#)).
- Fix import error on Python 3.5.0 and 3.5.1 (#267 by [@gongqingkui](#)).
- Fix parsing of unquoted values containing several adjacent space or tab characters (#277 by [@bbc2](#), review by [@x-yuri](#)).

0.14.0 - 2020-07-03

Changed

- Privilege definition in file over the environment in variable expansion (#256 by [@elbehery95](#)).

Fixed

- Improve error message for when file isn't found (#245 by [@snobu](#)).
- Use HTTPS URL in package meta data (#251 by [@ekohl](#)).

0.13.0 - 2020-04-16

Added

- Add support for a Bash-like default value in variable expansion (#248 by [@bbc2](#)).

0.12.0 - 2020-02-28

Changed

- Use current working directory to find `.env` when bundled by PyInstaller (#213 by [@gergelyk](#)).

Fixed

- Fix escaping of quoted values written by `set_key` (#236 by [@bbc2](#)).
- Fix `dotenv run` crashing on environment variables without values (#237 by [@yannham](#)).
- Remove warning when last line is empty (#238 by [@bbc2](#)).

0.11.0 - 2020-02-07

Added

- Add `interpolate` argument to `load_dotenv` and `dotenv_values` to disable interpolation (#232 by [@ulyssessouza](#)).

Changed

- Use logging instead of warnings (#231 by [@bbc2](#)).

Fixed

- Fix installation in non-UTF-8 environments (#225 by [@altendky](#)).
- Fix PyPI classifiers (#228 by [@bbc2](#)).

0.10.5 - 2020-01-19

Fixed

- Fix handling of malformed lines and lines without a value (#222 by [@bbc2](#)):
 - Don't print warning when key has no value.
 - Reject more malformed lines (e.g. "A: B", "a='b',c").
- Fix handling of lines with just a comment (#224 by [@bbc2](#)).

0.10.4 - 2020-01-17

Added

- Make typing optional (#179 by [@techalchemy](#)).
- Print a warning on malformed line (#211 by [@bbc2](#)).
- Support keys without a value (#220 by [@ulyssessouza](#)).

0.10.3

- Improve interactive mode detection ([@andrewsmith](#)) (#183).
- Refactor parser to fix parsing inconsistencies ([@bbc2](#)) (#170).
 - Interpret escapes as control characters only in double-quoted strings.
 - Interpret `#` as start of comment only if preceded by whitespace.

0.10.2

- Add type hints and expose them to users ([@qnighy](#)) (#172)

- `load_dotenv` and `dotenv_values` now accept an `encoding` parameter, defaults to `None` ([@theskumar](#)) ([@earlbread](#))([#161])
- Fix `str/unicode` inconsistency in Python 2: values are always `str` now. ([@bbc2](#))([#121])
- Fix Unicode error in Python 2, introduced in 0.10.0. ([@bbc2](#))([#176])

0.10.1

- Fix parsing of variable without a value ([@asyncee](#)) ([@bbc2](#))([#158])

0.10.0

- Add support for UTF-8 in unquoted values ([@bbc2](#)) ([#148])
- Add support for trailing comments ([@bbc2](#))([#148])
- Add backslashes support in values ([@bbc2](#))([#148])
- Add support for newlines in values ([@bbc2](#))([#148])
- Force environment variables to `str` with Python2 on Windows ([@greyli](#))
- Drop Python 3.3 support ([@greyli](#))
- Fix `stderr/-out/-in` redirection ([@venthur](#))

0.9.0

- Add `--version` parameter to cli ([@venthur](#))
- Enable loading from current directory ([@cjauvin](#))
- Add 'dotenv run' command for calling arbitrary shell script with `.env` ([@venthur](#))

0.8.1

- Add tests for docs ([@Flimm](#))
- Make 'cli' support optional. Use `pip install python-dotenv[cli]`. ([@theskumar](#))

0.8.0

- `set_key` and `unset_key` only modified the affected file instead of parsing and re-writing file, this causes comments and other file intact as it is.
- Add support for `export` prefix in the line.
- Internal refractoring ([@theskumar](#))
- Allow `load_dotenv` and `dotenv_values` to work with `StringIO()` ([@alanjds](#))([@theskumar](#))([#78](#))

0.7.1

- Remove hard dependency on iPython ([@theskumar](#))

0.7.0

- Add support to override system environment variable via `.env`. ([@milonimrod](#)) ([#63](#))
- Disable `".env not found"` warning by default ([@maxkoryukov](#)) ([#57](#))

0.6.5

- Add support for special characters `\`. ([@pjona](#)) ([#60](#))

0.6.4

- Fix issue with single quotes ([@Flimm](#)) ([#52](#))

0.6.3

- Handle unicode exception in `setup.py` ([#46](#))

0.6.2

- Fix `dotenv list` command ([@ticosax](#))
- Add iPython Support ([@tillahoffmann](#))

0.6.0

- Drop support for Python 2.6

- Handle escaped characters and newlines in quoted values. (Thanks [@iameugenejo](#))
- Remove any spaces around unquoted key/value. (Thanks [@paulochf](#))
- Added POSIX variable expansion. (Thanks [@hugochinchilla](#))

0.5.1

- Fix `find_dotenv` - it now start search from the file where this function is called from.

0.5.0

- Add `find_dotenv` method that will try to find a `.env` file. (Thanks [@isms](#))

0.4.0

- cli: Added `-q/--quote` option to control the behaviour of quotes around values in `.env`. (Thanks [@hugochinchilla](#)).
- Improved test coverage.



Помощь

Установка пакетов [↗](#)
Загрузка пакетов [↗](#)
Руководство пользователя [↗](#)
Project name retention [↗](#)
Часто задаваемые вопросы

О PyPI

PyPI в Twitter [↗](#)
Панель мониторинга
инфраструктуры [↗](#)
Статистика
Логотипы и товарные знаки
Наши спонсоры

Внесение вклада в PyPI

Использование PyPI

- Программные ошибки и обратная связь

Внесение вклада на GitHub'е

Перевод PyPI

Спонсор PyPI

Вклад в разработку
- Нормы поведения

Сообщить о проблеме безопасности

Политика конфиденциальности

Условия пользования

Политика допустимого использования

Статус: Service Under Maintenance

Разработано и поддерживается сообщество Python'а для сообщества Python'а.
Пожертвуйте сегодня!

PyPI", "Python Package Index" и логотипы блоков являются зарегистрированными
товарными знаками Python Software Foundation.

© 2023 Python Software Foundation
Карта сайта

Переключиться на настольную версию

English

español

français

日本語

português (Brasil)

українська

Ελληνικά

Deutsch

中文 (简体)

中文 (繁體)

> русский

עברית

esperanto



AWS
Cloud computing
and Security
Sponsor

Datadog
Monitoring

Fastly
CDN

Google
Download
Analytics

Microsoft
PSF Sponsor

Pingdom
Monitoring

Sentry
Error logging

StatusPage
Status page