
debuggie

Выпуск

BARS Group

22 May 2014

1	Установка	3
2	Подключение	5
3	Запись отладочной информации	7
4	Просмотр файла дампа отладочной информации	9
5	debuggie.api	11
6	debuggie.urls	13
7	debuggie.views	15
8	Indices and tables	17
	Содержание модулей Python	19

Contents: Debuggie - это набор инструментов для сбора и анализа отладочной информации об функционировании удалённой системы.

Набор включает в себя:

Middleware, собирающую отладочную информацию Viewie - CLI-утилиту, предназначенную для просмотра дампов отладочной информации API для отладочного вывода информации из кода отлаживаемых приложений

Установка

Пакет устанавливается стандартным способом:

```
pip install debuggie
```

Подключение

```
# =====  
# settings.py  
# =====  
  
INSTALLED_APPS = (  
    ...,  
    "debuggie",  
    ...  
)  
  
MIDDLEWARE_CLASSES = (  
    ...,  
    'django.contrib.sessions.middleware.SessionMiddleware',  
    'django.contrib.auth.middleware.AuthenticationMiddleware',  
    "debuggie.api.Debuggie",  
    ...  
)  
  
# функция, предоставляющая информацию о системе  
DEBUGGIE_SYSTEM_INFO_CALLBACK = "package.module.function"  
  
# =====  
# urls.py  
# =====  
urlpatterns = patterns(  
    ...,  
    (r'^debug/', include('debuggie.urls')),  
    ...  
)
```

Предупреждение: Debuggie middleware должна добавляться после SessionMiddleware и AuthenticationMiddleware, т.к. зависит от них!

Запись отладочной информации

1. Пользователь аутентифицируется в системе,
2. Указанием в адресной строке переходит на url `<some_host>/debug`
3. На открытой странице можно включать/выключать режим отладки, очистить дамп отладочной информации или загрузить его в виде `.txt`-файла. При этом ссылки на загрузку и очистку дампа появятся после того, как пользователь:
 - (a) Включит отладочный режим
 - (b) Совершит в системе какие либо действия
 - (c) Обновит страницу отладочного режима.

Просмотр файла дампа отладочной информации

Информация просматривается в терминале командой `viewie <dump_file.txt>` Программа просмотра работает в интерактивном режиме и содержит встроенную справку

debuggie.api

`class debuggie.api.Debuggie`
Middleware, ведущая запись дампов отладочной информации

`debuggie.api.clear_records(request)`
Сбрасывает дамп отладочной информации

`debuggie.api.get_default_system_info()`
Возвращает некоторые сведения о системе

`debuggie.api.get_download_url(request)`
Возвращает url для загрузки дампа, если файл уже существует. В противном случае возвращает None

`debuggie.api.get_fname(request)`
Возвращает имя файла, в который происходит запись дампа

`debuggie.api.get_header(request)`
Возвращает заголовок блока отладочной информации

`debuggie.api.get_status(request)`
Возвращает признак: “режим отладки включен/выключен”

`debuggie.api.get_sys_info()`
Возвращает дамп информации о системе на основе данных, возвращаемых функцией, указанной в `settings.DEBUGGIE_SYSTEM_INFO_CALLBACK`

Важно: Информация получается путем вызова ф-ции один раз, и затем кэшируется!

`debuggie.api.tell(data, record_type='tell')`
Добавляет блоков отладочной информации новый блок, если режим отладки включен, в противном случае ничего не пишется

`debuggie.api.toggle_status(request)`
Включает/выключает режим отладки для текущего пользователя

`debuggie.urls`

debuggie.views

`debuggie.views.debug_clear(request)`

Очистка списка событий

`debuggie.views.debug_status(request)`

Вывод текущего состояния логгера

`debuggie.views.debug_toggle(request)`

Переключение состояния логгера

Indices and tables

- *genindex*
- *modindex*
- *search*

d

`debuggie`, 1
`debuggie.api`, 9
`debuggie.urls`, 11
`debuggie.views`, 13