

## Программирование Python.COM объекты в Django Наумов Р. В.

*Наумов Родион Владимирович / Naumov Rodion Vladimirovich - студент,  
кафедра информационных технологий,  
Институт математики и информатики  
Северо-Восточный федеральный университет имени М. К. Аммосова, г. Якутск*

**Аннотация:** в статье предлагается начальная часть программирования на Python с помощью фреймворка Django. Создание виртуального окружения и установка библиотек. Пример работы на ActivePython и COM объектов.

**Ключевые слова:** python, django, COM.

В настоящее время Python является одним из самых востребованных языков программирования. Python прост в использовании, так же является объектно-ориентированным языком программирования.

Одним из самых лучших инструментов для разработки веб-сервисов на языке программирования Python является фреймворк Django. В библиотеке Django есть также готовые шаблоны для регистрации, авторизации, администраторская часть создается автоматически, что позволяет сэкономить достаточное количество времени [1].

Плюсы использования Django:

- Наличие администраторской части;
- Выбор готовых шаблонов;
- Обширная документация;
- Достаточно высокая скорость работы.

COM – объекты – это стандарт заданный Microsoft. Благодаря COM объектов возможно открытие word, excel, power point и др. [2].

COM – объекты входят в дистрибутив ActivePython.

С помощью Django возможно реализовать веб-сервис с вызовом Word или Excel, запись имеющихся данных, сохранение.

Для начала работы необходима установка виртуального окружения (virtualenv), непосредственная установка фреймворка Django [3].

Создание виртуального окружения на Python27:

C:\myproject>C:\Python27\Scripts\virtualenv.exe venv

Первая часть C:\myproject> - текущий каталог.

Вторая часть C:\Python27\Scripts\virtualenv.exe – путь, где установлен виртуальное окружение(устанавливается виртуальное окружение: c:\Python27\Scripts\pip install virtualenv).

Третья часть venv – имя виртуального окружения.

После установка библиотек Django с помощью команды pip install Django

Создание функции в views.py на рис. 1.

```

def Comobject(request):
    pythoncom.CoInitialize()
    try:
        word = client.Dispatch("Word.Application")
        doc = word.Documents.Open("C:\\myproject\\Test.rtf")
        fields = doc.Fields
        f1 = iter(fields)
        f2 = next(f1)
        f2.Result.Text = '123'
        doc.Save()
        word.Application.Quit()
    finally:
        pythoncom.CoUninitialize()

```

*Рис. 1. Функция в views.py*

Для работы с COM-объектами необходимо использование win32com.client.

В этой функции import win32com.client as client.

word = client.Dispatch("Word.Application") – работа с word, если необходима работа с другими офисными приложениями вместо слова word задать название приложения(например Excel.Application).

doc = word.Documents.Open("C:\\myproject\\Test.rtf") – открытие word, в данном случае файл формата rtf(Rich Text format)

В установленном rtf файле имеется поле MergeField, надлежит использовать макросы, встроенные в Microsoft Office

После открытия word документа находит первый макрос и заменяет данный макрос на число 123. Сохраняет изменения и выходит с word, необходимо освободить word их процессов.

В этом примере был рассмотрен простой способ вызова и записи в COM- объектах в Python на Django. В более углубленном изучении возможна автоматизация ввода данных в word, excel и д.р. на разработанном сайте.

Одним из самых больших минусов использования win32com является невозможность развернуть на linux сервере. Что является большой проблемой, т.к. win32com был разработан специально для Windows.

### *Литература*

1. Django By Example / Под ред. Antonio Mele. – Packt Publishing, 2015. 24 с.
2. Learning Django Web Development/ Под ред. Sanjeev Jaiswai, Ratan Kumar - Packt Publishing, 2015. 43 с.
3. Python for Data Analysis / Под ред. Wes McKinney – O'Reilly Media. Inc, 2012. 125 с.