Реализация первых методов api.

№ урока:	10	**Курс:**	Python Practice

Средства обучения:	Интерпретатор Python, virtualenv, текстовый редактор

Обзор, цель и назначение урока

Цель урока: расширить функционал созданного web приложения, добавить методы api для возврата информации о текущих курсах валют системы.

Изучив материал данного занятия, учащийся:

- Научится реализовывать методы api для создаваемых web сервисов.
- Научится работать с динамическими url во Flask приложениях.
- Научится обрабатывать GET запросы и получать аргументы GET запроса во Flask приложениях.
- Закрепит навыки работы с ORM библиотекой peewee.
- Закрепит навыки при работе с xml и json форматами данных.
- Закрепит навыки написания тестов для собственного кода.

Содержание урока

1. Краткое резюме, что мы уже имеем к данному момент и обзор дальнейших действий.
2. Добавление view-функции с динамическим url для api метода получения курсов валют.
3. Реализация контроллера с логикой обработки GET запроса, обработкой параметров GET запроса.
4. Создание класса базового контроллера, помещение общей логики в него - обработка запросов, запись логов, обработка ошибок.
5. Формирование ответа в формате xml и json, в зависимости от url запроса.
6. Написание и запуск тестов для проверки api с использованием библиотеки requests.

Резюме

- Каркас нашего приложения уже готов, есть уже даже страница с отображением курсов в виде html таблицы. Сегодня нам предстоит реализовать первый api метод в нашем приложении
- То есть в нашем web сервисе появится новый url, при обращении на который методом GET, клиент получит информацию о текущих курсах валют из базы. При этом, результат выполнения запроса может быть в формате json или xml. Также в качестве get аргументов могут быть переданы параметры from_currency, to_currency для уточнения курса валют, который интересует клиента. Необходимо предусмотреть обработку этих аргументов.
- Прежде всего нужно добавить новую view функцию в модуль views.py, указать, какой url будет соответствовать этой функции. Во Flask есть возможно указывать динамические url, то есть url с динамической частью, которая передается в качестве именованного аргумента во view функцию.
- Синтаксис следующий: можно добавить секцию <variable_name> в url. И тогда в сигнатуре, соответствующей view функции будет описан аргумент variable_name.
- Дополнительно можно указывать конвертер, для уточнения типа аргумента. Возможные типы: string (по умолчанию), int, float, path, uuid. Числовае типы int, float — только положительные числа.
- Для получения аргументов GET запроса во Flask приложении нужно воспользоваться параметром args объекта Flask запроса. Этот параметр возвращает объект типа

CyberBionic systematics
CyberBionic Systematics ® 2018
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine
Tel. 0 800 750 312
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com

werkzeug.datastructures.ImmutableMultiDict, основное отличие от обычных dict словарей — возможность добавлять несколько значений по одному ключу. Также ImmutableMultiDict, что следует из его названия, недоступен для редактирования, только для чтения значений и ключей.
- Для формирования ответа в формате json нужно воспользоваться функцией flask.json.jsonify. Для формирования ответа в формате xml нужно будет сформировать xml документ и поместить его в тело ответа с помощью функции flask.make_response
- Для тестирования будем использовать библиотеку requests и отправлять запросы на получение курсов по созданным url api.

Закрепление материала

- Как можно указать динамическую чать url во Flask приложении?
- Какие типы можно указать в динамической части url?
- Каким образом можно получить доступ к get параметрам url запроса во Flask приложении?
- В чем отличие метода get и getlist у объекта типа ImmutableMultiDict?
- Какой принцип ООП используется в контроллерах в коде проекта?

Дополнительное задание

Задание

Добавить обработку GET параметра updated при отправке запроса на получение курсов валют. Возвращать только те курсы, время обновления у которых больше updated. Сделать соответствующий тест.

Самостоятельная деятельность учащегося

Задание 1

Изменить xml формат ответа, возвращать список элементов xrate с атрибутами from, to, rate, updated. Сделать соответствующий тест.

Рекомендуемые ресурсы

http://flask.pocoo.org/docs/1.0/quickstart/#variable-rules
http://flask.pocoo.org/docs/1.0/api/#flask.Request.args
http://werkzeug.pocoo.org/docs/0.14/datastructures/#werkzeug.datastructures.ImmutableMultiDict
http://flask.pocoo.org/docs/1.0/api/#flask.json.jsonify

CyberBionic Systematics ® 2018
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine

Tel. 0 800 750 312
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com