

УДК 004.42(045)

# СОЗДАНИЕ ВЕБ-ПРИЛОЖЕНИЯ ДЛЯ РАБОТЫ С ФАЙЛАМИ JUPYTER NOTEBOOK В РАМКАХ УЧЕБНОГО ПРОЦЕССА

**Носко М. М.,**

студент факультета прикладной математики и информационных технологий,  
Финансовый университет, Москва, Россия  
maxnosko@mail.ru

**Аннотация.** В статье рассмотрены методика и программные инструменты, используемые в процессе обучения программированию на языке Python, который на сегодняшний день является одним из самых востребованных языков программирования в мире. Исследуется проблема, связанная с невозможностью просмотра файлов интерактивной оболочки Jupyter Notebook на мобильных устройствах, варианты решения данной проблемы, а также обоснована необходимость создания веб-приложения, решающего данную проблему. Вместе с необходимостью создания приложения в статье содержится описание создания данного приложения, включающее выбор средств для ее реализации, техническое и программное обеспечение, использованное во время разработки, описание алгоритмов работ. Приведен внешний вид интерфейса приложения, описано его применение, а также полезность данной программы для учебного процесса.

**Ключевые слова:** Python; Jupyter Notebook; обучение программированию; web-приложение для мобильных устройств

## CREATION OF THE WEB-APPLICATION FOR WORKING WITH JUPYTER NOTEBOOK FILES IN THE EDUCATIONAL PROCESS

**Nosko M.M.,**

student, Faculty of applied mathematics of information technologies,  
Financial University, Moscow, Russia  
maxnosko@mail.ru

**Abstract.** The article describes the methodology and software tools used in the process of programming training in Python, which is today one of the most popular programming languages in the world. The author investigated the problem of impossibility to view files of the Jupyter Notebook interactive shell on mobile devices. The author offers solutions to this problem, and also justified the need to create a web application that solves this problem. Together with the rationale for the creation of such an application, the article also contains a description of the process of creating this application, including the choice of means of its implementation, technical requirements and software used during the development, and a description of the algorithms. Further, I presented the appearance of the application interface, described its application, as well as the usefulness of this program for the educational process.

**Keywords:** Python; Jupyter Notebook; programming training; mobile web-application

Научный руководитель: **Макрушин С.В.**, кандидат экономических наук, доцент Департамента анализа данных, принятия решений и финансовых технологий, Финансовый университет, Москва, Россия.

Одним из главных направлений стратегических задач Российской Федерации на период до 2024 г. является развитие цифровой экономики и цифрового обучения. В настоящее время соответствующая целевая программа разрабатывается Правительством РФ.

Наш Финансовый университет при Правительстве РФ, очевидно, может и должен принять активное участие в реализации поставленной задачи.

В развитии цифровой экономики и цифрового обучения важную роль мог бы сыграть факультет прикладной математики и информационных технологий.

В статье предлагается разработанный и уже апробированный на практике способ совершенствования данного процесса цифрового обучения.

Изучение программирования на языке Python – одно из важнейших элементов цифрового обучения в нашем университете.

В процессе обучения программированию на языке Python используется интерактивная оболочка Jupyter Notebook (далее – JN), входящая в дистрибутив языка Python Anaconda.

Язык программирования Python был разработан Гвидо Ван Россумом (1956 г.р.) – нидерландским программистом, участвовавшим в создании языка ABC (языка программирования, предназначенного для обучения программированию) в 1980–1991 гг.

Язык Python – высокоуровневый и интерпретируемый язык программирования. Он поддерживает такие парадигмы программирования, как функциональная, объектно-ориентированная и многие другие [1, с. 13]. Это позволяет программисту при написании или представлении на естественном языке программы абстрагироваться от вычислительных процессов, которые будут производиться компьютером во время ее выполнения, и тратить большую часть времени на продумывание результата программы [2, с. 1]. Пример реализации на языке Python высокоуровневой концепции – возможность использовать внутренние методы сортировки структурированных данных языка с помощью использования стандартной функции вместо выбора и написания алгоритма сортировки с использованием стандартных операторов

сравнения и присваивания в самом тексте кода программы.

Так как язык Python является интерпретируемым, при выполнении программы используется специальная программа-интерпретатор, которая анализирует каждую команду, из которых состоит программный код, после чего выполняет ее.

Особенности синтаксиса языка Python:

- достаточное соответствие названий стандартных функций и ключевых слов на английском языке их обычному значению, не связанному с программированием;
- данный подход к именованию хорошо сбалансирован с небольшой, но способствующей однозначному пониманию длиной имен ключевых слов и функций;
- минималистичный синтаксис, который наделяет широкими возможностями специальные символы, но не затрудняет понимание написанной программы: значение специальных символов синтаксиса Python подобно их значению в обычных языках.

Данные особенности позволяют:

- сделать написание, чтение и редактирование программ на языке Python простым для программиста [3, с. 33];
- упростить процесс обучения на данном языке [4, с. 13].

Именно простота процесса обучения программированию на языке Python, сочетающаяся с его широкими возможностями и разнообразными областями применения, обладает актуальностью. Фактор актуальности очень важен, ведь он увеличивает ценность знаний, получаемых во время обучения. Хотя существуют языки, процесс обучения программированию на которых проще, чем на Python, но они не имеют актуальных на сегодняшний день модулей для решения задач, стоящих перед языками программирования. В качестве примера можно рассмотреть язык программирования Pascal, созданный для обучения [5, с. 44]. Будучи изначально не предназначенным для решения практических задач, на сегодняшний день он совершенно не актуален.

Вышеприведенные особенности языка Python позволяют использовать его, в том числе, для обучения программированию.

JN – это интерактивная оболочка языка программирования Python. Она содержит в себе

дополнительный функционал и программный синтаксис, а также алгоритмы для подсветки кода.

Использование JN упрощает освоение студентами языка Python с помощью наглядного и упрощенного отображения результата работы программы: в целях обучения программу можно разделить на необходимое количество частей — блоков и видеть результат выполнения каждого из них. Это повышает учебную эффективность выполняемых заданий, а также позволяет наглядно демонстрировать особенности языка студентам.

Система Jupyter Notebook имеет свой формат файлов (с расширением `ipynb`), отличный от обычных файлов исходного кода программ на языке Python (с расширением `py`). Это обусловлено необходимостью хранения в файлах системы JN дополнительной информации, например блоков с текстовым содержимым, отформатированным с использованием языка разметки Markdown, которые могут быть встроены в файл JN между блоками с кодом.

По причине того, что файлы системы JN не являются ни файлами исходного кода программ на языке Python, ни обычными текстовыми файлами, их невозможно открыть на мобильном устройстве, что затрудняет использование системы: безусловно, для выполнения файлов нужна непосредственно система JN, но отсутствие возможности просмотра файлов на мобильных устройствах — проблема, затрудняющая учебный процесс: во время лекций, которые проводятся не в компьютерных классах, студенты не имеют возможности просматривать файлы JN с учебными, в том числе с лекционными материалами на мобильных устройствах.

Таким образом, существует необходимость в создании приложения, которое облегчит процесс взаимодействия с файлами JN на различных устройствах. Это приложение должно как минимум позволять просматривать содержимое данных файлов.

Перед созданием приложения под названием WebTea было проведено изучение альтернативных вариантов решения данной проблемы.

Система JN осуществляет хранение файлов в формате JSON (текстовый формат хранения

структурированных данных<sup>1</sup>), т.е. это файлы, представляющие из себя систему строк, разделенных специальными символами, поэтому они могут быть открыты обычным текстовым редактором. Безусловно, просмотр и изменение файлов возможно с помощью обычных текстовых редакторов, но файлы содержат в себе слишком большое количество специальных символов, а также все строки исходного текста подвергаются экранированию (добавлению специальных символов перед некоторыми символами). Это означает, что от пользователя потребуется умение ориентироваться в системе хранения данных JSON, технические символы которой значительно затруднят чтение, а изменение будет сопровождаться риском сделать файл нечитаемым для системы JN, имеющей собственную структуру хранения блоков в JSON.

---

***Именно простота процесса обучения программированию на языке Python, сочетающаяся с его широкими возможностями и разнообразными областями применения, обладает актуальностью.***

---

Несмотря на то что существуют редакторы JSON для мобильных устройств, они способны открывать файлы JN и позволяют пользователю просматривать их содержимое немного удобнее, чем текстовые редакторы. Однако подобные программы не ориентированы на работу конкретно с файлами JN, что затрудняет взаимодействие с данными файлами. Редактирование, так же как и в случае с текстовыми редакторами, сопровождается риском сделать файл нечитаемым для системы JN.

Одним из альтернативных вариантов также является запуск системы JN непосредственно на мобильных устройствах. Единственным преимуществом данного способа можно считать возможность выполнения программ на мобильных устройствах. Однако есть ряд причин, из-за которых такой вариант нельзя назвать неподходящим решением поставленной задачи:

---

<sup>1</sup> Подробная информация о JSON. URL: <http://json.org/json-ru.html> (дата обращения: 23.07.2018).

- запуск системы JN невозможен без применения стороннего программного обеспечения<sup>2</sup>, так как запуск на мобильных устройствах не поддерживается официально<sup>3</sup>;
- такой способ запуска JN достаточно сложный: для различных мобильных платформ пользователю необходимо будет выполнить нетривиальные последовательности действий для запуска JN;
- необходимо иметь в виду, что интерфейс JN не адаптирован для мобильных устройств: достаточно большое количество функций для удобной работы с системой невозможно использовать без наличия физической клавиатуры (или ее сенсорного аналога), что затрудняет использование такого варианта;
- из-за различий между мобильными устройствами запуск JN на них не всегда возможен.

***Использование JN упрощает освоение студентами языка Python с помощью наглядного и упрощенного отображения результата работы программы: в целях обучения программу можно разделить на необходимое количество частей – блоков и видеть результат выполнения каждого из них.***

Следовательно, целесообразным является создание приложения, которое бы одинаково работало как на мобильных, так и на настольных устройствах. Другими словами, было необходимо найти подход, позволяющий создать кроссплатформенное приложение, которое сможет выполнять поставленную задачу: отображать содержимое файлов JN в привычном для пользователя JN виде.

В качестве данного подхода было решено использовать веб-интерфейс, т.е. создать веб-

приложение. Это было обосновано тем, что именно веб-интерфейс позволяет пользователю взаимодействовать с приложением, используя лишь браузер, который по умолчанию есть как на мобильных устройствах, так и на настольных. Это также делает взаимодействие с приложением одинаковым и простым на различных устройствах. Безусловно, такое взаимодействие требует наличия у пользователя Интернета, но это ограничение не существенно в том числе и из-за того, что на территории учебного заведения нет проблем с доступом к Интернету.

Возможности веб-приложения значительно шире, чем только просмотр файлов, поэтому было решено дополнить минимально необходимый функционал дополнительным – создать приложение, которое включает в свой функционал как просмотр файлов JN, так и их хранение, а также обмен файлами между пользователями.

Для создания приложения были выбраны языки программирования PHP и JavaScript (далее – JS), которые широко используются вместе для создания подобных по структуре клиент-серверных систем.

Язык программирования PHP был разработан датским программистом Расмусом Лердорфом (1968 г.р.) в 1995 г. Данный язык используется для создания динамических веб-страниц [6, с. 16]. Он обладает весьма широкими возможностями, позволяющими ему быть серверной стороной веб-приложений. PHP выполняет процессы, связанные с обработкой и хранением данных, использующихся в веб-приложении, после чего передает клиентской стороне приложения веб-интерфейс, состоящий динамически в результате обработки большого количества информации:

- данных, переданных клиентской стороной (например, сведений для авторизации);
- сведений, хранящихся в базе данных;
- информации, которую можно получить в результате запроса к стороннему сетевому ресурсу.

Язык программирования JS был создан американским программистом Эйхом Бренданом (1961 г.р.) в 1995 г. JS чаще всего используется при создании динамических веб-страниц, так как он поддерживается ведущими браузерами (такими, как Opera, Google Chrome и Mozilla Firefox), в которых у данного языка есть достаточный уровень

<sup>2</sup> Пример запуска Jupyter Notebook на мобильном устройстве. URL: <http://www.leouieda.com/blog/scipy-on-android.html> (дата обращения: 23.07.2018).

<sup>3</sup> Список ОС, официально поддерживаемых JN. URL: <https://www.anaconda.com/download/> (дата обращения: 23.07.2018).





Рис. 1. Внешний вид системы просмотра файлов JN

Источник: веб-приложение WebTea.

интеграции для взаимодействия с элементами веб-страницы, а также для работы со сторонними сетевыми ресурсами. С целью защиты пользователей от вредоносного контента, загружаемого чаще всего из Интернета и выполняемого браузером, ведущие браузеры не позволяют JS-программе, выполняющейся на интернет-странице, оказывать влияние на компьютер пользователя. Это обусловлено тем, что процесс выполнения может быть инициирован пользователем таким простым действием, как нажатие мышью на гиперссылку, текст которой может не соответствовать ее реальному содержанию и последствиям нажатия.

Следовательно, программа на языке JS не может хранить важные данные на компьютере пользователя. Из-за этого JS идеально подходит для создания интерактивного интерфейса HTML-страницы [7, с. 23], каждый раз создающегося браузером и зависящего от данных, которые были сформированы сервером в каждом случае загрузки страницы.

HTML – это язык разметки веб-страниц, разработанный в 1993 г. Больше всего на сегодняшний день используется для создания

разметки страниц, состоящих из элементов, располагаемых на экране устройства браузером, исходя из особенностей физического оборудования, используемого пользователем. HTML позволяет передавать в виде текста (например, по протоколу HTTP) разметку, достаточную для построения веб-страницы, содержащую в себе изображения, интерактивные элементы (гиперссылки, текстовые поля, кнопки и т.д.) и текстовые элементы разного цвета, размера и шрифта. Для того чтобы элементы могли изменяться в рамках одной страницы, необходимо использовать дополнительные программные средства, например JS, так как HTML не позволяет изменять содержимое страницы (но позволяет перейти на другую страницу с другим содержимым [8, с. 8], что отчасти может создать интерактивный веб-интерфейс).

Для функционирования приложения был развернут Apache HTTP-сервер. Запросы к серверу осуществляются по статическому IP-адресу, куда ссылается соответствующее доменное имя.

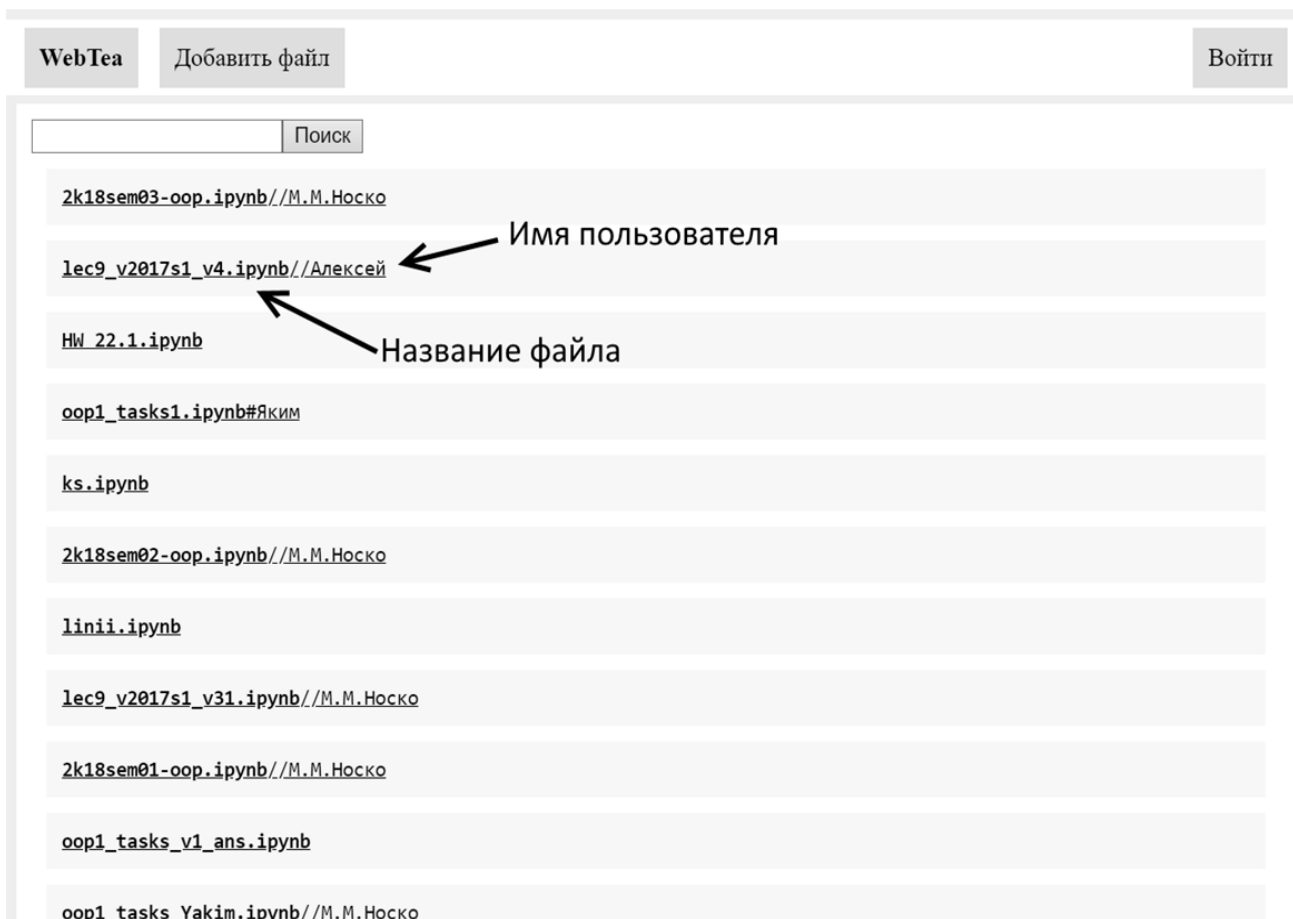


Рис. 2. Внешний вид списка загруженных файлов

Источник: веб-приложение WebTea.

В первую очередь была реализована система просмотра файлов JN в браузере: создана JS-программа, которая анализирует предоставленный пользователем через веб-интерфейс файл JN и составляющая его HTML представление для отображения в браузере. Для удобства пользователя HTML представление файла имеет внешний вид, аналогичный графическому интерфейсу взаимодействия с файлами, имеющими расширение ipynb непосредственно через Jupyter Notebook (рис. 1).

Система просмотра файлов получает их от JSON-базы данных, реализованной с помощью PHP алгоритмов, выполняющихся на сервере. Соответственно, файлы хранятся в текстовом виде в JSON-файле, и по требованию пользователя сервер формирует текстовый файл, идентичный первоначальному файлу JN, который принимается системой JN, при этом информация, которая изначально присутствовала в блоках JN, полностью сохраняется.

После этого был создан веб-интерфейс для взаимодействия пользователей с загруженными

ми в базу данных, расположенную на сервере, файлами. В приложение также была добавлена система авторизации, имеющая веб-интерфейс, что позволяет пользователям загружать файлы под своим именем и скрывать их от других пользователей. Их можно загрузить без авторизации, но файлы, не принадлежащие конкретному пользователю, могут быть удалены или изменены любым другим пользователем приложения, в том числе и не авторизованным.

Помимо этого, в приложение были добавлены функции переименования файлов, добавление к файлам коротких текстовых описаний-комментариев и редактирования файлов. Редактирование также доступно пользователям системы, если файлы принадлежат им или если файл был загружен неавторизованным пользователем (если пользователь не авторизован, то владельцем файла будет считаться любой неавторизованный пользователь).

Соответственно, только если у пользователя есть право на просмотр файла, он будет

ему показан. Файлы, скрытые от остальных пользователей, в списке файлов выделены дополнительным цветом.

В списке файлов пользователю предоставлена возможность поиска файлов по имени или по фамилии пользователя, загрузившего их (рис. 2).

Как и в любых клиент-серверных системах, в данном приложении система состоит из двух основных частей – клиентской и серверной: браузер пользователя – клиент – отправляет HTTP запросы к серверу.

На сервере выполняются PHP алгоритмы, которые работают с серверной базой данных и отвечают за авторизацию и внесение изменений в хранящиеся в базе данных файлы. Соответственно, все процессы, происходящие на сервере, скрыты от пользователя: пользователю сообщается только результат их работы. Пользователь лишь инициирует операции с помощью сетевого запроса и получает ответ от сервера также посредством HTTP. При этом сервер получает от клиентской стороны набор данных: адрес запрашиваемой PHP страницы, GET и POST данные, а также Cookies. Все эти данные передает серверу браузер пользователя с помощью HTTP.

Выполняемый сервером PHP-алгоритм формирует HTML-страницу, содержащую в себе только то содержимое, на доступ к которому имеет право пользователь, а также JS-код, предназначенный, соответственно, для выполнения браузером пользователя.

С помощью JS пользовательский интерфейс становится интерактивным: возможности JS позволяют динамически изменять HTML-страницу – веб-интерфейс взаимодействия с пользо-

вателем. Особенности построения интерфейса с помощью HTML позволяют адаптировать его для любых устройств, благодаря возможностям мобильных и настольных браузеров, а также взаимодействовать с приложением как с помощью физических мыши и клавиатуры, так и сенсорного экрана.

Авторизованный пользователь может хранить на сервере файлы системы JN, а также редактировать и удалять их, используя пользовательский интерфейс.

Приложение будет активно развиваться. Планируется создание функционала для создания файлов JN с возможностью совместного редактирования группами пользователей, разделение базы данных на разделы, доступ к которым смогут осуществлять установленные группы пользователей, и создание системы динамического формирования файлов JN из базы блоков.

Проблема, связанная с невозможностью просмотра файлов Jupyter Notebook на мобильных устройствах, была решена с помощью интеграции созданного веб-приложения в учебный процесс. Использование данного сервиса позволило улучшить эффективность обучения программированию на языке Python. Согласно статистике, которая автоматически собирается сервером, в среднем за неделю приложение открывают 20 раз.

Таким образом, суммируя всю приведенную выше информацию, можно сказать, что данное приложение приносит пользу – оказывает положительное влияние на учебный процесс, упрощая процесс коллективного взаимодействия с файлами JN.

### Список источников

1. Прохоренок Н.А. Python 3 и PyQt. Разработка приложений. СПб.; 2012. С. 13–14.
2. Lutz M. Python Pocket Reference, Fifth Edition. Sebastopol, CA; 2014. P. 1–2.
3. Lutz M. Learning Python, Fifth Edition. Sebastopol, CA. 2013. P. 33–34.
4. Саммерфильд М. Программирование на Python 3. Подробное руководство. СПб.; 2009. С. 13–15.
5. Грацианова Т.Ю. Программирование в примерах и задачах. М.; 2015. С. 43–44.
6. Маклаффин Б. PHP и MySQL. Исчерпывающее руководство. СПб.; 2013. С. 15–16.
7. Флэнаган Д. Javascript. Подробное руководство. 5-е изд. СПб.; 2008. 23 с.
8. Росс В.С. Создание сайтов: HTML, CSS, PHP, MySQL. Учеб. пособие. Ч. 1. М.; 2010. С. 7–12.