

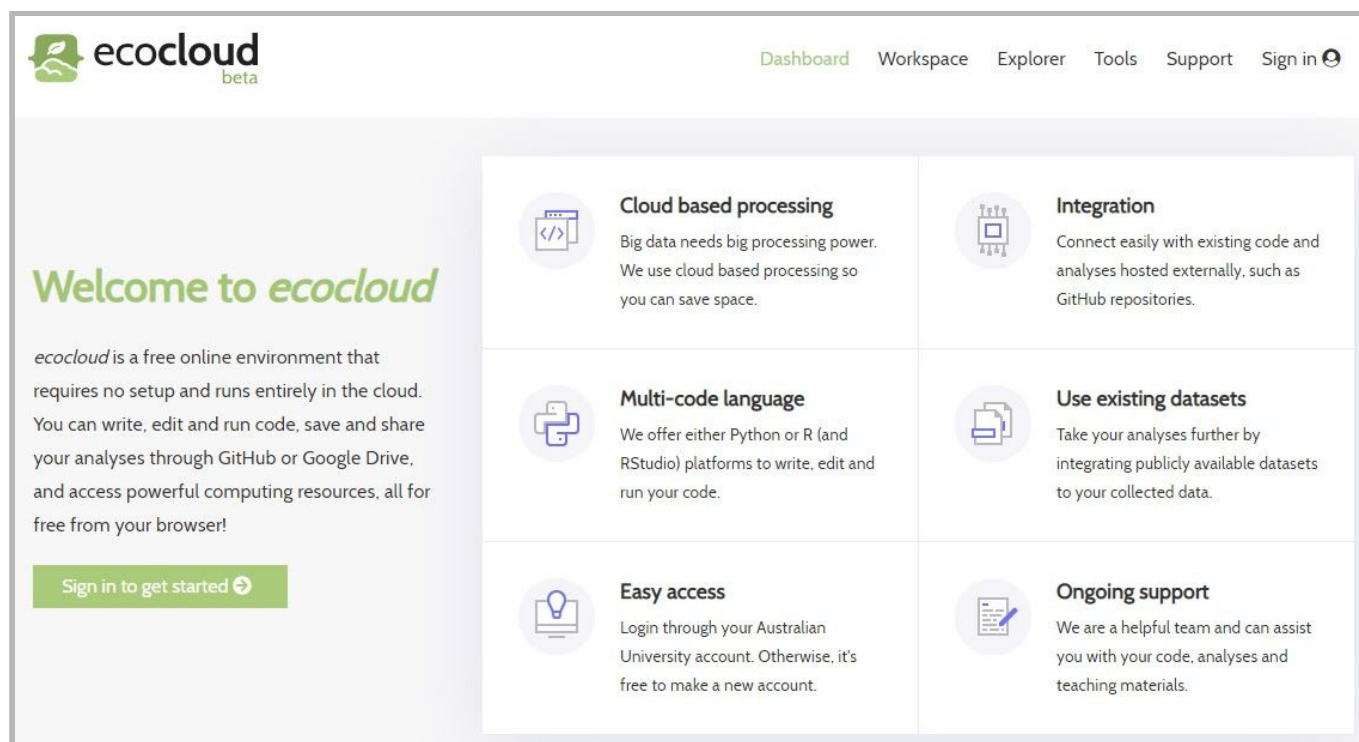
Ecocloud - виртуальная среда для работы с R и Python

Для начала работы потребуется только интернет-браузер. Рекомендуется использовать **Google Chrome** или **Mozilla FireFox**. Не рекомендуется **Microsoft Internet Explorer**, некоторые функции веб-приложений в нем могут работать не корректно, особенно в более ранних версиях.

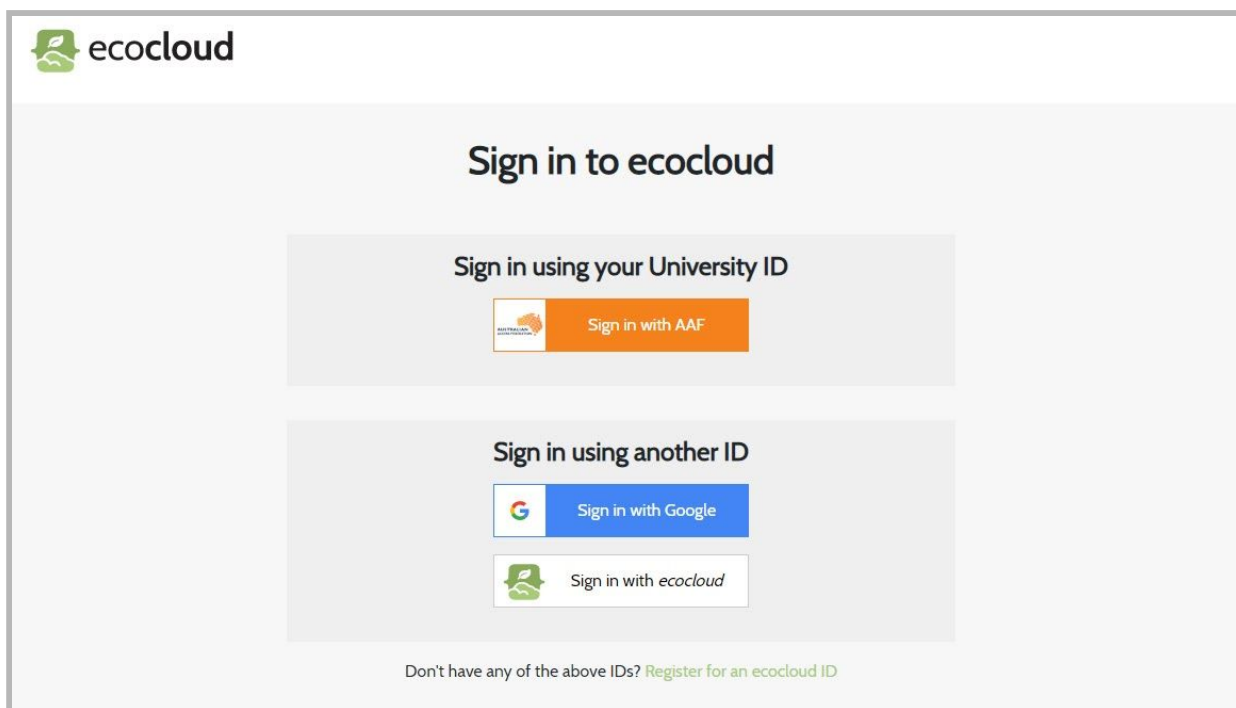
Также может понадобиться:

1. Учетная запись на портале GBIF.
[Краткая инструкция как зарегистрироваться](#) - на сайте GBIF.ru
2. Простой текстовый редактор, например, для просмотра CSV-файлов и небольших правок в них. Рекомендуется использовать [Notepad++](#)
3. Геоинформационная система [QGIS](#) для визуализации и анализа полученных пространственных данных. Желательно установить модули (плагины) веб-картографии *AusMap* и *QuickMapServices*.

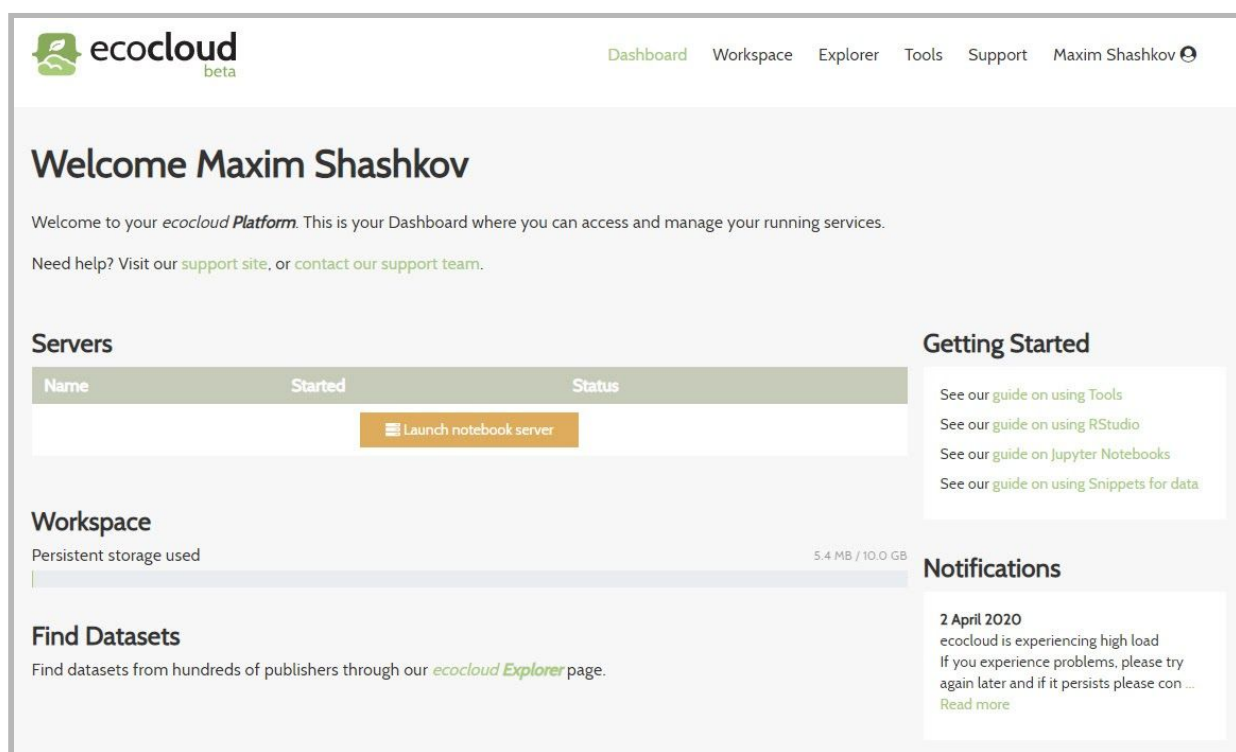
Облачная платформа **ecocloud** находится по адресу <https://ecocloud.org.au> и для того, чтобы начать с ней работать, необходимо авторизоваться или создать учётную запись (зарегистрироваться). Для этого переходим по ссылке "**REGISTER / LOG IN**", попадаем на страницу с описанием основных функций и особенностей данной системы, далее нажимаем кнопку "**Sign in to get started**"



Для входа можно использовать уже существующую учетную запись **Google**, если таковой у вас нет, то регистрируемся, перейдя по ссылке "**Register for an ecocloud ID**" (более мелким шрифтом на сером фоне под кнопками)



Авторизовавшись, вы попадаете в основное меню. Два раздела, с которыми мы в основном будем работать - это **Dashboard** и **Workspace**. Первый (Dashboard) служит для запуска виртуального сервера, при помощи которого будет происходить ваша работа с данными, второй раздел (Workspace) - для работы с файлами, содержащими наборы данных и скрипты (программный код), служащие для их обработки.



Раздел **Tools** по большей части дублирует **Dashboard**. В разделе **Support** есть краткие инструкции и руководства по работе с системой. Раздел **Explore** содержит ресурсы, на которых можно найти разнообразные наборы данных.

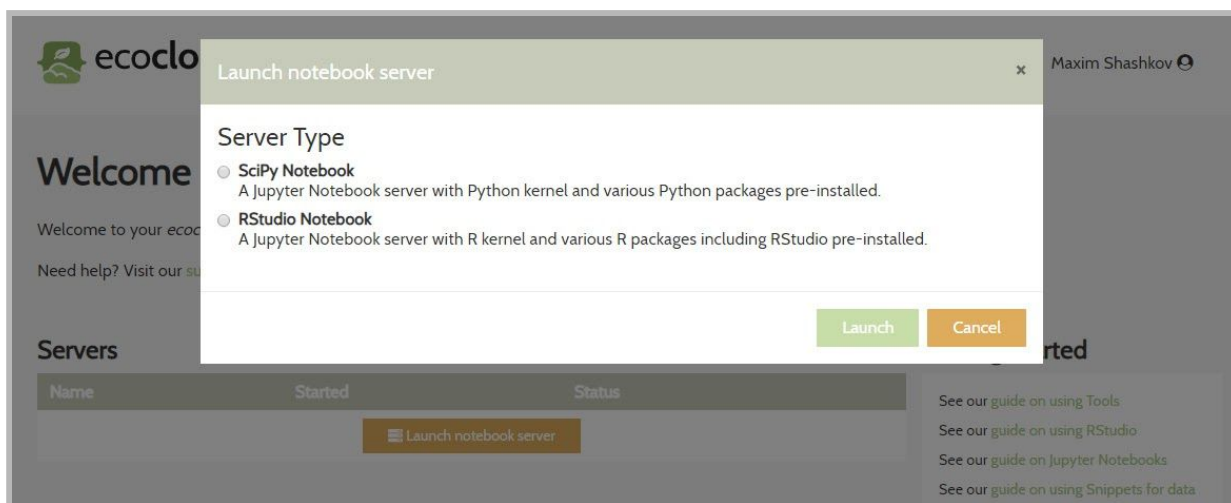
Начнем знакомство с системой **ecocloud** с раздела **Workspace**. По своей сути - это файловый менеджер, при помощи которого вы можете загружать свои файлы в облако и сортировать их по директориям в предоставляемом дисковом пространстве. Затем их можно использовать для вычислений в виртуальном сервере. Под ваше рабочее пространство отводится 10 Gb дискового хранилища сервиса **ecocloud**. Для начала работы этого будет более чем достаточно, так как таблицы и скрипты для их обработки обычно много места не занимают. Для удобства здесь можно создавать директории (папки) как на своём компьютере.

Есть один важный нюанс: непосредственно во время работы со средой **ecocloud** вы будете создавать новые файлы и редактировать уже существующие, **НО** все новые файлы и изменения в уже существующих файлах будут существовать только во время текущего сеанса работы с системой. Чтобы сохранить результаты, важные файлы надо сохранить на свой компьютер, а затем отправить в свой аккаунт через **Workspace**. Немного непривычно может быть на первый взгляд, но такая система позволит вам избежать накопления множества временных файлов и неактуальных версий файлов.

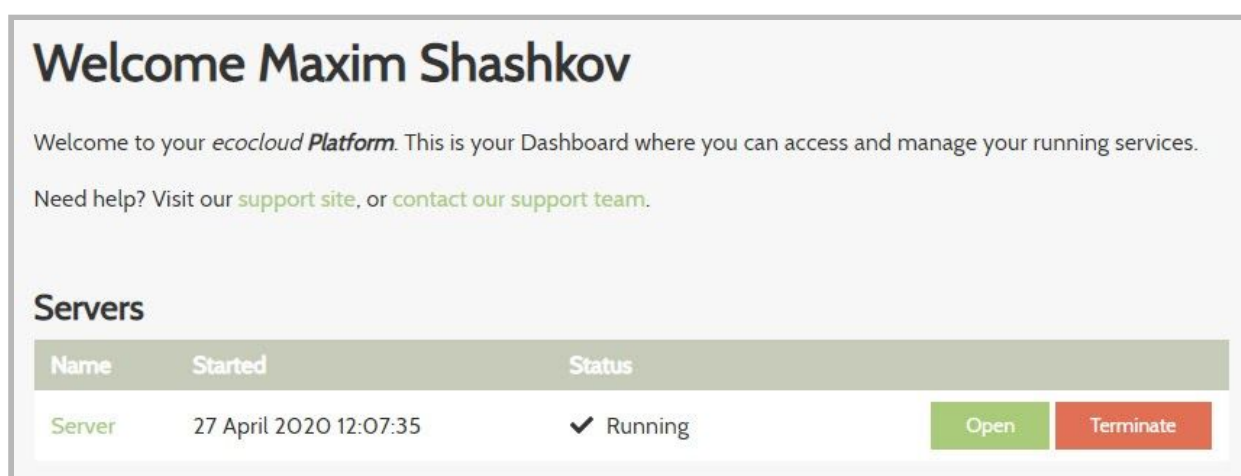
Теперь перейдём в раздел **Dashboard**. Есть два ключевых момента в работе системы **ecocloud**, первый - во время сеанса работы для вас запускается небольшой виртуальный сервер, имеющий все необходимые настройки для работы со статистическим пакетом **R** или языком программирования **Python**. Это избавляет от необходимости устанавливать на свой компьютер и настраивать дополнительное ПО. Но, не исключено, что для вычислений в виртуальной среде вам потребуется загрузить и установить какие-то дополнительные пакеты для языка **R** или библиотеки для **Python**, сделать это можно практически так же, как на локальном компьютере.

Еще одна ключевая особенность, а точнее основная для данного сервиса - программный код организован в виде специального формата [Jupyter Notebook](#) или "блокнот Jupyter", также называемого "вычислительным блокнотом". В нем фрагменты программного кода чередуются с подробными пояснениями и комментариями, которые могут содержать иллюстрации, формулы и таблицы. Отдельные блоки кода можно запускать все сразу или пошагово, последовательно или в произвольном порядке. Вы сразу же увидите результат выполнения каждого блока кода и при необходимости сможете что-то в нем изменить и запустить нужный участок кода заново. Подобная технология очень эффективна для обучения, совместной разработки, организации расчетов для исследований, оформления интерактивных отчетов и многих других задач. Файлы с вычислительными блокнотами имеют расширение *.ipynb.

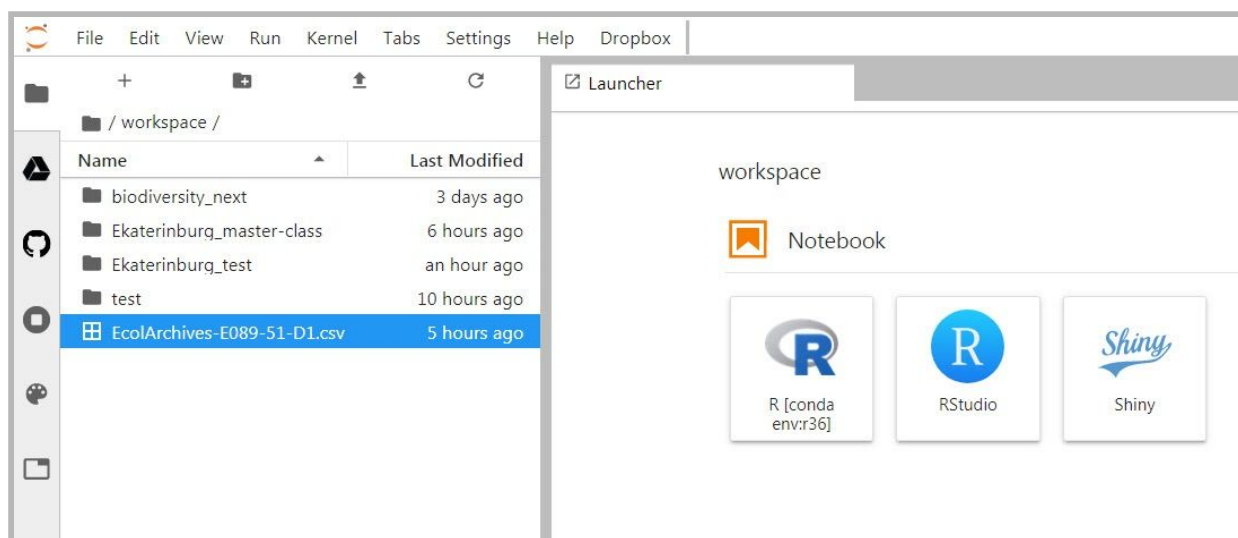
Итак, запустим сервер - кнопка "**Launch notebook server**". В **ecocloud** есть два варианта конфигурации запуска: для среды статистической обработки данных **R** и для языка программирования **Python**, одновременно можно запустить только один вариант, для первого занятия выбираем **RStudio Notebook**. В среде такого виртуального компьютера и будет происходить работа с вычислительными блокнотами, без него блокнот - это просто структурированный текстовый файл (формата обмена данными JSON, кому интересно).



После того, как сервер с заданной конфигурацией будет запущен, ссылка на него появится под заголовком "**Servers**". Чтобы начать работу с ним, нажмите кнопку "**Open**".



Для того, чтобы создать новый блокнот, надо нажать кнопку "**R [conda env: r36]**", но также можно найти и открыть уже существующий, при этом не только из своего рабочего пространства, но и из внешних ресурсов. Для этого в левой части окна есть небольшая панель инструментов (вертикальный ряд значков).



Назначение кнопок боковой панели инструментов, по порядку:

File Browser - доступ к вашему рабочему пространству с файлами. Важно помнить, что все новые файлы и новые версии уже имеющихся, созданные при работе с виртуальным сервером, будут существовать только во время его работы (такие файлы отмечены зеленой точкой слева от названия) и пропадут, когда сервер будет остановлен. Поэтому все результаты, которые нужны, надо сохранить на свой компьютер (или на флеш-диск), а затем загрузить в свою учётную запись через раздел *Workspace*.

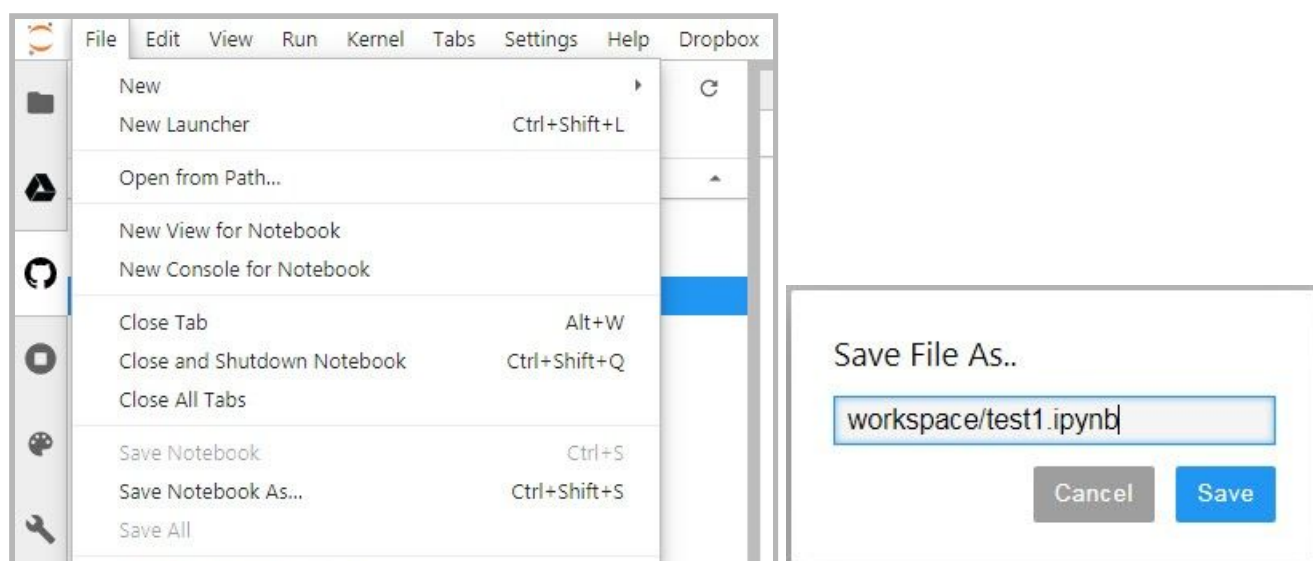
Google Drive - здесь вы можете подключить своё файловое хранилище на Google и брать нужные файлы оттуда. Файлы в процессе работы можно редактировать, но их также необходимо сохранять на локальном компьютере, так как возможность изменять что-то на диске Google через интерфейс **ecocloud** не реализована.

Browse GitHub - доступ к крупнейшему хостингу IT-проектов. Здесь можно найти множество вычислительных блокнотов и учебных курсов на их основе, но большинство из них будет на английском языке. В начале здесь будет ссылка на учебные примеры от разработчиков сервиса "**ausecocloud/training**". Чтобы получить доступ к определенному репозиторию надо ввести его адрес в виде "*пользователь/репозиторий*", например, материалы нашего курса находятся здесь: "**MaxCarabus/SDMworkshop**". Ссылку надо вводить в адресной строке панели инструментов. Если адрес введен правильно, вы увидите список файлов данного репозитория. Файл с блокнотом для нашего первого занятия называется "**Lesson_1.ipynb**".

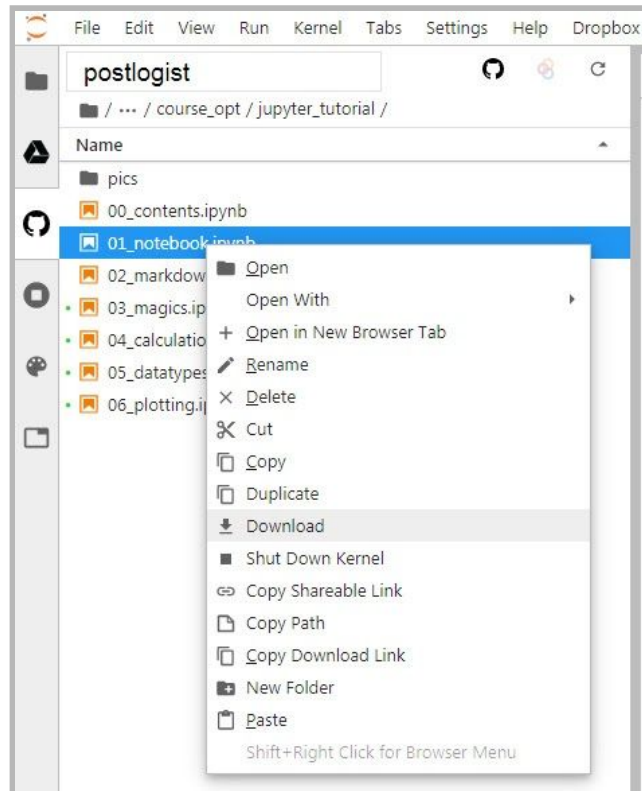
Файлы, полученные с **GitHub**, можно редактировать, но изменения прямо в этом разделе сохранить нельзя. Чтобы продолжить работу с блокнотом, загруженным из **GitHub**, его надо сохранить в своём рабочем пространстве как временный файл (первый вариант), или через локальный диск и загрузку файлов в разделе **Workspace** как постоянный.

В первом случае через основное меню **File -> Save Notebook As ...**

и в появившемся небольшом окне указать путь и имя файла в своём рабочем пространстве:

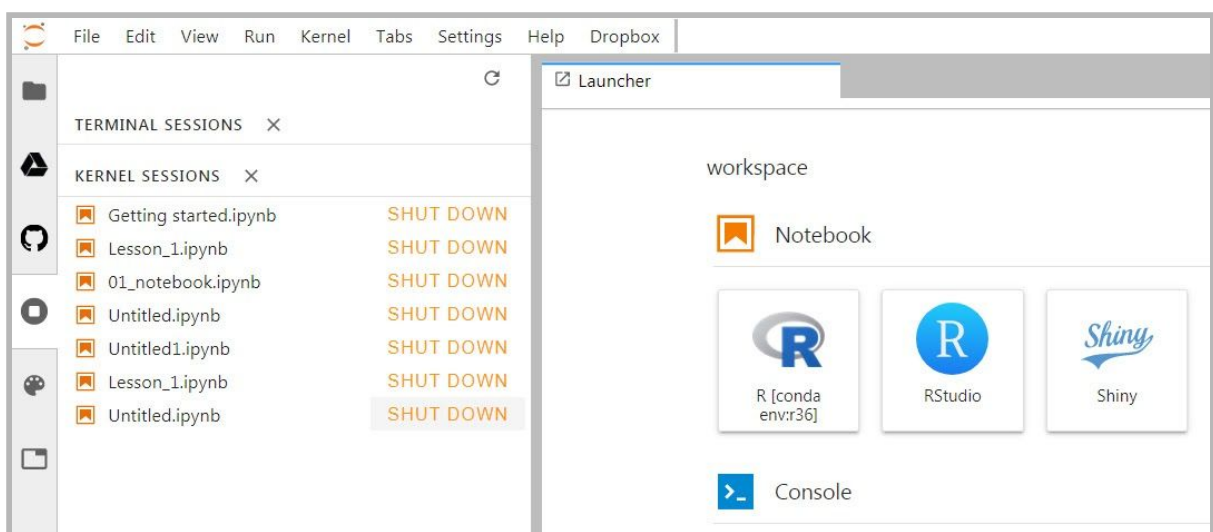


Чтобы сохранить файл на свой компьютер, выберите команду **"Download"** из контекстного меню, затем его можно отправить в своё файловое хранилище **ecocloud** через раздел **Workspace**.



Для работы с каждым блокнотом виртуальный сервер запускает отдельное "ядро" (kernel), это "движок", который по вашей команде выполняет отдельные ячейки вычислительного блокнота и хранит результаты вычислений (переменные).

Следующий инструмент **Running Terminals and Kernels** показывает вкладки с запущенными блокнотами. Как понимаете, можно запускать сразу несколько блокнотов и чтобы перейти к нужному, надо нажать на его название. Если блокнот сейчас не нужен, его можно закрыть, нажав на надпись **"SHUT DOWN"**.



Инструмент **"Commands"** открывает вкладку со всеми командами, которые используются в сервисе Jupyter, как для запуска программного кода, так и для работы с файлами и редактирования блокнотов.

Инструмент **"Open Tabs"** показывает список открытых вкладок. Перейдя на вкладку **"Launcher"** вы сможете не только создать новый вычислительный блокнот, но и открыть web-интерфейс среды разработки **RStudio** или привычную консоль **R**.

После того, как вы запустили виртуальный сервер, он будет работать даже если вы выйдете из своей учётной записи. Это позволит вернуться через какое-то время к вычислениям в открытых блокнотах, но надо иметь в виду, что запущенный виртуальный сервер имеет ограничение по времени работы и через 24 часа будет остановлен автоматически и все файлы созданные в процессе сеанса работы будут потеряны. Поэтому своевременно сохраняйте результаты вычислений и актуальные версии блокнотов на диск своего компьютера.

После того, как работа завершена и все необходимые файлы сохранены на локальном диске, можно остановить виртуальный сервер, нажав на кнопку **"Terminate"** в разделе **Dashboard**, таким образом вы уменьшите нагрузку на системные ресурсы сервиса.

В **GitHub** есть репозиторий проекта Jupyter. В нем есть [большой список ресурсов](#) с вычислительными блокнотами по самой разнообразной тематике. На видео-хостинге **YouTube** вы можете посмотреть два небольших видео-ролика, автор которых рассказывает об [основах вычислений в Jupyter notebook](#) и также о [редактировании блокнотов](#). При желании, можно использовать эти материалы для самостоятельного освоения.

*старший научный сотрудник
Института математических проблем биологии -
филиала Института прикладной математики им. М.В. Келдыша,
Максим Шашков*

