# Установка необходимого программного обеспечения и библиотек на Mac OS X

На Mac OS обычно уже установлен Python. Более того, если ваша операционная система обновлена, компьютер сейчас использует Python 3.X, при желании можно установить самые последние версии Python, протестировать беты и так далее. На бетах всё, естественно, будет работать косо и криво, поэтому без особой уверенности в себе беты, а тем более гаммы и прочие пре-релизы лучше не трогать.

Тем не менее, если Python вдруг не установлен, вот, что нужно сделать:

* Переходим на Python.org
* Выбираем Downloads
* Обычно сайт сам определяет, какой у нас браузер и ОСь, качаем предложенный пакет, открываем .pkg, который мы только что скачали, следуем инструкции

Для установки целого огромного количества программ можно использовать такую штуку, как Homebrew или MacPorts. Что это за штуки? Это установщик (менеджер) пакетов и приложений, если вы имели дело с Linux, brew может вам напомнить apt-get. Итак, для начала установим менеджер (обычно оба ставить не рекомендуют, так как они могут конфликтовать, но если вы нормально чувствуете себя, обращаясь с системным администрированием, вы можете ставить оба):

* /usr/bin/ruby -e "$(curl -fsSL <https://raw.githubusercontent.com/Homebrew/install/master/install>)"
* Скачиваем подходящую под ОСь версию менеджера MacPorts: <https://github.com/macports/macports-base/releases/tag/v2.6.2>

Для корректной работы не только этих менеджеров, но часто и Python и прочего-прочего требуется установить x-code command line tools, в терминале вводим следующую команду:

* xcode-select –install

В некоторых случаях может потребоваться перезапустить компьютер, иногда этого не требуется. Переходим к самому Питону и к установщикам пакетов, conda/pip. В первую очередь проверим какая версия Python используется нашим компьютером, для этого введём в терминал следующую команду:

* Python -V

Затем проверим pip, который помогает нам устанавливать библиотеки:

* pip -V

Теперь установим anaconda, скачав соответствующий пакет на сайте: <https://www.anaconda.com/distribution/#macos> Раскрываем .pkg файл, следуем инструкции, устанавливаем. Анаконда это дистрибутив, там сразу же содержится много библиотек и всяких полезностей. Дистрибутив может напомнить дополнение для игры или игру, продаваемую с набором дополнений. Если вы хотите работать именно с Anaconda, к ней также можно скачать графическую среду разработки. Есть версия PyCharm, собранная специально для Anaconda (<https://www.jetbrains.com/pycharm/promo/anaconda/>), есть достаточно знаменитый инструмент Jupyter Notebook (<https://jupyter.org>), обычный PyCharm можно использовать в научном режиме (Scientific Mode), не скачивая специальную версию для Anaconda. В науке также часто используют ПО Orange (<https://orange.biolab.si>).

Итак, вернёмся к Anaconda:

A screenshot of a cell phone

Description automatically generated

После установки мы сможем использовать менеджер Conda, он отличается от Pip совсем немного. Есть лишь несколько библиотек весьма редких, специфичных и высокопрофессиональных, которые требуют либо один менеджер, либо другой, например RDKit, используется в химинформатике. Скорее всего вам хватит либо одного, либо другого менеджера.

Обратите внимание, когда мы набираем в терминале команду: ‘pip -V’, мы получаем не только версию pip, но и путь к pip, а также путь к используемой версии Python, например: ‘pip 19.2.3 from /Library/Frameworks/Python.framework/Versions/3.8/lib/python3.8/site-packages/pip (python 3.8)’; стоит запомнить, где «лежит» Python, при настройке IDE нам пригодится этот путь. Это путь «интерпретатора», условно «читальщика» кода. Нам надо будет его указать, чтобы запускать скрипты в PyCharm.

Так как c PyCharm всё довольно просто, сосредоточимся на установке Jupyter. На самом деле, пользоваться им можно и онлайн, из браузера: <https://jupyter.org/try> Но, если копия всё же нужна на своём компьютере:

* conda install -c conda-forge notebook
* pip install notebook

Достаточно использовать одну команду на выбор.

\*\*\* Иногда conda так просто не работает на Mac OS, в новых версиях используется терминал zsh, я не буду вдаваться в то, что это такое, но зато покажу, как прописать путь к conda, чтобы её легко можно было вызвать в zsh:

* atom ~/.zshrc

Atom это простой текстовый редактор, можно использовать другие текстовые редакторы, такие как vim, ultraedit, sublimetext и другие, их очень много, у каждого есть свои предпочтения. То есть вместо слова «atom» вы пишите любое другое слово, которое вызовет избранный текстовый редактор.

В файл, который мы только что открыли, тут или там, мы пишем следующее:

* source ~/.bash\_profile

Теперь надо сохранить файл (можно закрыть). Вернёмся в терминал:

* source ~/.zshrc

Исполняем. Теперь, если вы введёте, что-то вроде:

* conda -V

В ответ вы получите версию conda, которую теперь использует ваш компьютер\*\*\*

При установке Anaconda в папке Applications появляется приложение Anaconda-Navigator, отсюда также можно начать веб-версию Jupyter, чтобы не путаться, просто запустите Jupyter Lab, в браузере вы увидите опции для работы.

A screenshot of a cell phone

Description automatically generatedТеперь снова вернёмся к PyCharm. После установки и выбора необходимых опций, запустите ПО, зайдите во вкладку View:

A screenshot of text

Description automatically generated

В виде вам надо выбрать инструменты:

A screenshot of a cell phone

Description automatically generated

Как только появится галочка, вы увидите сверху гаечный ключ:

A screenshot of a computer screen

Description automatically generatedНажмите на гаечный ключ:

A screenshot of a computer

Description automatically generatedПод названием проекта выберите тот самый интерпретатор-читальщик. Если вы настраиваете в первый раз свой проект, вам надо будет добавить интерпретатор. Вот где нам понадобится путь к питону!

A screenshot of a computer

Description automatically generated

В появившемся списке выбираем «Показать всё», нажимаем:

A screenshot of a computer

Description automatically generated

Выбираем «+» в левом нижнем углу. Здесь мы можем выбрать и указать путь к нужной версии Python, опций много, я советую просто с ними поиграть:

A screenshot of a cell phone

Description automatically generatedДля установки новых библиотек/пакетов, нам нужно снова вернуться к настройкам (гаечный ключ, название проекта, интерпретатор и плюсик):

A screenshot of a computer

Description automatically generated

А теперь выбираем необходимый пакет:

A screenshot of a cell phone

Description automatically generated

Устанавливаем библиотеку/пакет. Иногда могут возникнуть ошибки. Изучите отчёт. Может не хватать какого-то компонента. Стоит установить: cython, wheel, flask; они могут быть необходимы для корректной работы NumPy. Изучите отчёт по ошибке, скопируйте его, просмотрите в текстовом редакторе. Скорее всего с такой проблемой кто-то уже сталкивался, решения зачастую можно найти на StackOverFlow. Библиотеки, которые разрабатывают отдельные энтузиасты, обычно имеют либо свою страничку (верхний скрин-шот, справа, текст в синем), либо такую же на github. В случае ошибки, можно спросить автора в соответствующем раз деле.

Почти всё. Теперь посмотрим, как работает Scientific Mode. Я пока только импортировала NumPy, но среда разработки уже предлагает использовать научный режим:

A screenshot of a computer screen

Description automatically generated

См. правый нижний угол. Научный режим позволяет мне посмотреть полученный массив (матрицу) без вывода результата в консоль:

A screenshot of a cell phone

Description automatically generated

А теперь посмотрим то же, но в обычном режиме, в консоли ничего нет:

A screenshot of a computer screen

Description automatically generated

Результат появляется только тогда, когда я добавляю print():

A screenshot of a computer

Description automatically generatedСупер коротко, как установить необходимые библиотеки через терминал. Так как Anaconda, это научный дистрибутив, большая часть необходимого находится уже там, чтобы проверить, что есть, в терминале вводим команду:

* conda list

Скорее всего там уже будут необходимые нам библиотеки: pandas, NumPy, MatPlotlib, но при необходимости установим пакеты так:

* conda install numpy
* conda install pandas
* conda install matplolib

Если используется pip, делаем так:

* pip install pandas numpy matplotlib

При одновременной установке пакетов не разделяем названия запятыми, только пробелами!