

Библиотеки

Здравствуйтесь. В этом видео мы расскажем вам о том, какие библиотеки лучше изучать с детьми и как это сделать.

Вступление

Вначале стоит рассказать о том, что вообще такое библиотеки и для чего они нужны. Тут можно привести аналогию с конструктором (1.png), где наша программа - это конструктор, а библиотека - отдельные детали, из которых мы собираем нашу программу. Так же можно упомянуть, что немалую популярность Python приобрёл именно благодаря своим библиотекам и большому комьюнити, которое поддерживает и разрабатывает новые.

Встроенные библиотеки

Начать изучение библиотек лучше со встроенных. Говорим о том как они импортируются в скрипт (import) (2.png), и сразу же можно привести в пример библиотеку math (2.png), показать несколько функций из неё и рассказать каким образом их можно использовать. Так же на примере библиотеки math можно показать работу с документацией по библиотекам - функции dir(3.png) и help(4.png).

Далее можно рассказать о структурах библиотек, о том что они могут иметь некоторую вложенность и состоять из нескольких подмодулей, предназначенных для своих задач. Тут нужно упомянуть о том, что импортировать можно не всю библиотеку целиком, а только отдельные её элементы (from <модуль> import <элемент>)(5.png), а так же элементы можно переименовывать (from <модуль> import <элемент> as <своё название>)(6.png).

Следующей можно показать библиотеку random, её основные функции (randint, randrange, shuffle, choice и т.д.)(7.png). С использованием этих функций можно сделать несколько простых игр (от подкидывания монетки и бросания костей до перемешивания карт).

Далее можно рассказать о модуле datetime, он часто используется во многих приложениях(8.png). Рассказать о его структуре, и об основных функциях. Тут обязательно стоит рассказать о том, что datetime имеет собственный тип данных, это не обычный строчный формат(9.png). Так же хорошо рассказать о функции deltatime (10.png), а так же о строчном представлении типа datetime(11.png). Конечно рассказ нужно закрепить практической работой.

Можно использовать и библиотеку turtle (12.png). Вообще говоря можно воспользоваться ею раньше, она крайне хороша для наглядной демонстрации простейших конструкций языка - условий, циклов и т.д.

Можно упомянуть библиотеку os и поработать с файлами, это будет полезно для рассмотрения следующей библиотеки (13.png).

Затем можно рассмотреть tkinter (14.png). Стоит сказать о том что конечно python не самый лучший язык для создания десктопных приложений, но и на нём можно писать крутые программы. Тут ребята могут дать волю фантазии и придумать что-то своё. Уже тут можно начинать проектную деятельность, разбиться на команды и начать разработку.

Загружаемые библиотеки

Можно показать детям ресурс pypi.org(15.png) и провести аналогию с PlayMarket или AppStore - люди создают библиотеки и выкладывают в общий доступ, откуда каждый может взять нужную ему и пользоваться ей для своих задач.

Начать можно в библиотеке numpy(16.png), на примере которой нужно показать как удобно и быстро она работает с массивами, особенно с многомерными.

Далее нужно перейти к библиотеке Pillow (17.png). Во-первых, она подразумевает использование numpy, который значительно упрощает и ускоряет обработку изображений, во-вторых эта библиотека позволяет понять ребятам как вообще устроены растровые изображения. И стоит начать с этого - рассказать что изображение - это двумерный список (таблица, матрица) пикселей, изменяя которые мы можем изменять и изображение. Тут же стоит упомянуть о модели RGB.

Затем можно перейти к библиотеке pygame (18.png). Она вызывает пожалуй наибольший интерес у детей. Это большой раздел и здесь много о чем можно рассказать, начиная с того как работает игровой цикл и заканчивая физикой и анимацией. Эта библиотека так же очень хорошо подходит для проектной деятельности благодаря своей наглядности и, опять же, живому интересу со стороны детей.

Дальше можно переходить к web(19.png). Это, пожалуй, самый большой раздел python, в том числе и потому что наиболее часто он используется именно в этой отрасли.

Начать тут стоит с библиотеки requests(20.png). Но сначала нужно хотя бы в общих чертах рассказать об HTTP-запросах, о том как они работают и как формируются. Тут конечно нужно рассказать так же о json и библиотеке для работы с этим форматом, а так же закрепить это на работе с какой-нибудь API, например по получению текущего курса валют, погоды или чего-то ещё.

Какой web может быть без HTML и CSS (21.png)? Хорошо, если ребята знают что это такое и с чем это едят, в обратном же случае придется им рассказать, хотя бы в общих чертах.

В качестве фреймворка лучше взять Flask (22.png), ведь в отличие от Django он не такой нагруженный, но с загружаемыми инструментами позволяет делать web-сервисы с очень широким функционалом.