В файлі Technical Details, де я коротко описувала всі технічні характеристики і можливості API Udacity, я уже наводила два конкретних способи працювати з вибраними даними. Один з них був спосіб, представлений в самій документації вибраного API, а другий – спосіб, який був поданий як приклад використання програми оболонки для python ([Python Wrapper for Udacity](https://github.com/tylucaskelley/udacity-api-python)). Якраз цей другий спосіб я зараз розпишу детальніше.

Передусім, ця програма-оболонка складається з двох частин:

* [Catalog](https://github.com/tylucaskelley/udacity-api-python#catalog) – дозволяє працювати з курсами
* [User](https://github.com/tylucaskelley/udacity-api-python#user) – безпосередньо працює з конкретним користувачем, для чого треба зайти в свій акаунт

У своєму проекті я буду використовувати лише першу частину – роботу з каталогом наявних курсів.

Щоб встановити бібліотеку роботи з оболонкою, я викликала в терміналі команду pip install udacity. Після цього в будь-якому своєму модулі можна буде зробити import udacity і можна буде працювати з API за допомогою програми-оболонки.

Так як каталог – це безпосередньо та частина, яку я буду використовувати, то я її опишу першою, а наприкінці приведу приклади можливостей роботи з користувачем.

**Catalog**

Каталог використовується для пошуку та сортування курсів за їх певними властивостями та особливостями. Там є багато зручних функцій, які полегшать мою роботу з даними.

Щоб дізнатися більше про кожен метод і що він повертає, можна викликати команду pydoc udacity.Catalog.

Приклад того, як можна працювати з каталогом курсів:

import udacity

c = udacity.Catalog()

tracks = c.tracks()

track\_names = [t['name'] for t in tracks]

web\_dev\_teachers = c.instructors('cs253')

nd001\_description = c.degree('nd001')['expected\_learning']

**User**

Робота з користувачем заключається в можливості переглядати курси і прогрес конкретного користувача.

Щоб дізнатися більше про кожен метод і що він повертає, можна викликати команду pydoc udacity.User.

Приклад того, як можна працювати з користувачем:

import udacity

user = udacity.User('email@example.com', 'password123')

name = user.name()

# print out quiz completion rate in each course

for course in user.enrollments():

prog = user.progress(course)

print('Course: ' + prog['title'])

print('\t' + str(prog['quizzes\_completed']) + '/'

+ str(prog['quiz\_count']) + ' quizzes completed')