



Yalantis Education

Python school

Дедлайн виконання —
до **14.12.21** включно

Тестове завдання складається з 2-х частин:

1. Перевірка рівня англійської мови.

[Зареєструйтесь на сайті](#) та вкажіть прізвище, ім'я та електронну адресу так, як ви вказали в реєстраційній формі на сайті.

Вам необхідно відповісти на 15 запитань, ліміт часу - 10 хвилин.

Важливо! Ви можете пройти тестування лише один раз.

2. Практичне завдання

Розробіть REST API для парку машин з водіями

Створіть наступні таблиці (моделі) - Driver + Vehicle

Driver:

- + id: int
- + first_name: str
- + last_name: str
- + created_at
- + updated_at

Vehicle

- + id: int
- + driver_id: FK to Driver
- + make: str
- + model: str
- + plate_number: str - format example "AA 1234 OO"
- + created_at
- + updated_at

Створіть перелік відкритих (без аутентифікацій) endpoint'ів для наступних операцій:

Driver:

- + GET /drivers/driver/ - вивід списку водіїв
- + GET /drivers/driver/?created_at__gte=10-11-2021 - вивід списку водіїв, які створені після 10-11-2021
- + GET /drivers/driver/?created_at__lte=16-11-2021 - вивід списку водіїв, котрі створені до 16-11-2021

- + GET /drivers/driver/<driver_id>/ - отримання інформації по певному водію
- + POST /drivers/driver/ - створення нового водія
- + UPDATE /drivers/driver/<driver_id>/ - редагування водія
- + DELETE /drivers/driver/<driver_id>/ - видалення водія

Vehicle:

- + GET /vehicles/vehicle/ - вивід списку машин
- + GET /vehicles/vehicle/?with_drivers=yes - вивід списку машин з водіями
- + GET /vehicles/vehicle/?with_drivers=no - вивід списку машин без водіїв

- + GET /vehicles/vehicle/<vehicle_id> - отримання інформації по певній машині
- + POST /vehicles/vehicle/ - створення нової машини
- + UPDATE /vehicles/vehicle/<vehicle_id>/ - редагування машини
- + POST /vehicles/set_driver/<vehicle_id>/ - садимо водія в машину / висаджуємо водія з машини
- + DELETE /vehicles/vehicle/<vehicle_id>/ - видалення машини

Вимоги до виконання:

- + README.md - короткий опис проекту - must have
- + Інструкція по розгортанню SETUP.md - must have
- + Список залежностей requirements.txt - must have
- + Python version >= 3.9.0
- + Frameworks - Django/Flask
- + Використання сторонніх бібліотек не заборонено - головне знати міру :-)
- + Database - SQLite
- + Github/Bitbucket/GitLab

P.S

- + Формат виводу date и datetime - "%d/%m/%Y" и "%d/%m/%Y %H:%M:%S"
- + Написання тестів будет плюсом
- + flake8/mypy/black/isort буде плюсом, але не впливає на підсумкову оцінку
- + Django Only - додавання Django Admin Panel буде плюсом, але не впливає на підсумкову оцінку

P.P.S

- + Інформація повинна передаватись/відображатись в форматі json - ніяких форм
- + Завдання зроблене з допомогою форм не зараховується - вчимося читати ТЗ
- + Завдання зроблене з закритими (з аутентифікацією) endpoints не зараховується - вчимося читати ТЗ
- + Завдання зроблене без SETUP.md також не зараховується - думаємо про наступні покоління :-)
- + Завдання зроблене без requirements.txt дивним чином не зараховується - так буває :-)

Після виконання завдання, надішліть посилання на репозиторій куратору школи на пошту education@yalantis.net до 14.12.2021 включно та зазначте тему листа "{Прізвище} Тестове завдання для Yalantis Python School".

Зверніть увагу! Фідбек по тестовому ми відправимо вам до 24.12.21 включно.

Ми залишаємо за собою право відмовитись давати докладний фідбек.