

Podział filmów - przygotowanie do utworzenia scenariuszy LO Marki

1. 08.03 - lekcja 0 (wprowadzenie w temat, testy kompetencji etc.) - najlepiej na koniec dnia
2. 15.03 - lekcje 1 i 2 (2h blok), drugi blok tego dnia jest na inne tematy wedle Twojego uznania (jeśli uznasz za stosowne, może to być dokończenie zajęć z wcześniejszego bloku albo zajęcia w szkole zamiast pracy domowej)
3. 22.03 - lekcje 3 i 4 (2h blok), drugi blok tego dnia jest na inne tematy wedle Twojego uznania
4. 29.03 - lekcja 5 (1 lekcja lub 2h blok) - uczniowie otrzymują ostatnią dawkę wiedzy, na kolejnej lekcji (jeśli się na to zgodzisz) w bloku kończą swoje projekty i przesyłają je Tobie (pliki/linki), drugi blok tego dnia jest na inne tematy wedle Twojego uznania
5. 05.04 - lekcja 6, omówienie wykonanych zadań i test kompetencji oraz ankieta oceniająca całość zajęć (na potrzeby rozliczenia projektu przed grantodawcą), pozostałe zajęcia tego dnia to powrót do normalnego trybu pracy w ramach zajęć szkolnych.

Lekcja w szkole (1 i 2):

1. Konfiguracja IDE PyCharm, tworzenie konta w GitHub.com a01
2. Przygotowujemy środowisko `venv` dla lokalnego projektu a02
3. Plik `requirements.txt` - zewnętrzne moduły, własne pliki `py` w projekcie a03
4. Minimalny program z wykorzystaniem PySimpleGUI a04
5. PySimpleGui - dokumentacja, przykłady użycia a05
6. Podstawowe typy danych w Python, zmienne a06
7. Typy zaawansowane: listy, słowniki a07
8. Importowanie z zewnętrznych modułów a08
9. Pętla `for` i listy a09
10. Instrukcja warunkowa `if ... else ...` a10
11. Wyświetlamy informację. a11
12. Rozpakowywanie tupli - pythonizm. a12
13. Pętla `while True` - sterowanie programem PySimpleGUI. a13
14. Skąd będziemy czerpać grafiki? Pixabay, Freepik i Flaticon. (b01)
15. Uruchamiamy edytor tekstów - format strony. b02
16. Uruchamiamy edytor grafiki - otwieramy przykładowy plik graficzny. b03
17. Uruchamiamy edytor HTML - podstawowa strona z szablonu. b04
18. Edytor HTML: Bootstrap - <https://getbootstrap.com/> b05

Praca samodzielna (1 i 2):

1. VirtualBox w Windows i jak importować maszynę `OVA` - aby pracować niezależnie od szkoły b06
2. Edytor tekstów: listy numerowane i nienumerowane b11
3. Edytor grafiki: warstwy i dodanie elementu b12
4. Edytor HTML: containers b13
5. Edytor tekstów: nagłówki i stopka b07
6. Edytor tekstów: style i spis treści b08

7. Edytor grafiki: zmiana rozmiaru i zapis XCF b09
8. Edytor HTML: różne znaczniki `meta` b10
9. Dodajemy elementy przycisków. a14
10. Dodajemy wyświetlanie obrazków. a15
11. Sposoby wprowadzania danych. a16
12. Poznajemy sterowanie. a17
13. PySimpleGui - tworzymy prosty program okienkowy - 1 (layout, listy) a18
14. Wyświetlanie większej ilości danych (output) a19
15. Edytor tekstów: zrzut zawartości okna aplikacji i dodanie do tekstu b14
16. Edytor grafiki: warstwy i dodanie tekstu. b15
17. Edytor HTML: różne elementy na stronie (display, images, listy). b16

Lekcja w szkole (3 i 4):

1. Poznajemy `Python Console` w PyCharm + Wykorzystujemy `requirements.txt` i instalujemy niezbędne elementy: `requests` a20
2. Wykonujemy `request.get()` - z serwisu <https://fastapi.jurkiewicz.tech/> pokazujemy odczytane dane a21
3. `JSON` i słowniki w Python a22
4. Użycie pętli `for` dla pokazania elementów słownika z serwisu <https://fastapi.jurkiewicz.tech/> a23
5. Listy jako elementy słowników a24
6. Słowniki jako elementy słowników a25
7. Edytor grafiki: eksport obrazu jako PNG b17
8. Edytor tekstów: eksport dokumentu do formatu PDF b18
9. Edytor HTML: różne elementy na stronie (navbar, listy, image). b19
10. Replikacja projektu z GitHub do PyCharm (open via VCS) i dodanie lokalnego venv (Add interpreter) i w `requirements.txt` `Install all packages` a39
11. Dodanie do repozytorium pracy z aplikacją PySimpleGUI i `Commit/Push` a40

Praca samodzielna (3 i 4):

1. Sprawdzamy dokumentację dla przykładowych API: a26/ a27/ a28
 1. <https://aviationstack.com/documentation>
 2. <https://numverify.com/documentation>
 3. <https://wttr.in/:help>
2. Generowanie API_KEY dla wybranego projektu (Aviationstack) - <https://aviationstack.com/signup/free> a29
3. Generowanie API_KEY dla wybranego projektu (Numerify) - <https://numverify.com/documentation> a30

Lekcja w szkole (5):

1. Definiowanie funkcji w Python. a34
2. Funkcje i zasięg zmiennych w Python. a35
3. Aktualizowanie wartości dla kluczy słowników a43
4. Tworzenie słowników i dodawania do nich elementów a44
5. Testujemy dostęp do danych API (aviationstack) a36
6. Testujemy dostęp do danych API (numverify) a37

7. Testujemy dostęp do danych API (wttr.in) a38
8. Poznajemy kody odpowiedzi API: poprawnych i błędnych a31 / a32
9. tworzymy własne repozytorium, pamiętamy o `.gitignore`, `README.md` oraz licencji a33
10. Sprawdzenie działania — skrypt odczytujący i prezentujący wybrane dane a41
11. Wysłanie projektu do serwisu GitHub a42

Praca samodzielna (5):

1. Tworzenie pełnej aplikacji
2. Tworzenie docelowego interfejsu aplikacji
3. Przygotowanie końcowe dokumentacji i strony w HTML