




Agenda kursu PythON: Academy

 Kurs **PythON: Academy** to kilkaset lekcji, dzięki którym w prosty i skuteczny sposób nauczysz się Pythona - **od pierwszej linii kodu do pierwszej pracy jako programista.**

Jeżeli masz pytania dotyczące kursu - **pisz śmiało:**
online@infoshareacademy.com

Podsumowanie kursu w pigułce

Kurs	Liczba lekcji	Liczba modułów	Czas trwania
PythON: Zrozumieć Programowanie	84	7	9,5 godziny
PythON: Początek	200	9	23,5 godziny
PythON: Wtajemniczenie	319	23	33 godziny
SUMA:	603	39	66 godzin

Spis treści

[Podsumowanie kursu w pigułce](#)

[Spis treści](#)

[PythON: Zrozumieć Programowanie](#)

[Moduł 1 - Wprowadzenie i przygotowanie środowiska](#)

[Moduł 2 - Dogadać się z Pythonem](#)

[Moduł 3 - Podstawowe typy zmiennych](#)

[Moduł 4 - Instrukcje warunkowe i operatory logiczne](#)

[Moduł 5 - Pętle](#)

[Moduł 6 - Funkcje](#)

[Moduł 7 - Podsumowanie kursu "Zrozumieć Programowanie"](#)

[PythON: Początek](#)

[Moduł 1 - Niezbędnik Adepta](#)

[Moduł 2 - Programowanie obiektowe - część 1](#)

[Moduł 3 - Programowanie obiektowe - część 2](#)

[Moduł 4 - Wbudowane struktury danych](#)

[Moduł 5 - Środowisko i biblioteki](#)

[Moduł 6 - Obsługa błędów i pliki](#)

[Moduł 7 - Scraping \[BONUS\]](#)

[Moduł 8 - Certyfikacja Python Institute \(przygotowanie do certyfikatu PECP-30-02\)](#)

[Moduł 9 - Test końcowy](#)

[PythON: Wtajemniczenie](#)

[Moduł 1 - Niezbędnik Adepta](#)

[Moduł 2 - Funkcje](#)

[Moduł 3 - Data i czas](#)

[Moduł 4 - Datetime, interwały czasowe i formatowanie](#)

[Moduł 5 - Strefy czasowe](#)

[Moduł 6 - Podstawy wyrażeń regularnych](#)

[Moduł 7 - Type hint](#)

[Moduł 8 - Abstrakcja, interfejs i Protocol](#)

[Moduł 9 - Iterator i generator](#)

[Moduł 10 - Context Manager](#)

[Moduł 11 - Moduł Collections - część 1](#)

[Moduł 12 - Moduł Collections - część 2](#)

[Moduł 13 - Dekoratory](#)

[Moduł 14 - Metaklasy](#)

[Moduł 15 - Wybrane wzorce projektowe](#)

[Moduł 16 - Testy - część 1](#)

[Moduł 17 - Testy - część 2](#)

[Moduł 18 - Współbieżność - część 1](#)

[Moduł 19 - Współbieżność - część 2](#)

[Moduł 20 - Współbieżność - część 3](#)

[Moduł 21 - Warsztat programistyczny: Git i workflow](#)

[Moduł 22 - Warsztat programistyczny: workflow i CI](#)

[Moduł 23 - Podsumowanie i test końcowy](#)

PythON: Zrozumieć Programowanie

▼ Moduł 1 - Wprowadzenie i przygotowanie środowiska

1. Dla kogo jest kurs "Zrozumieć Programowanie"?
2. Dlaczego warto zrozumieć programowanie?
3. Plan kursu i sposób nauki
4. Czym jest Python?
5. Przygotowanie środowiska pracy
6. Przygotowanie środowiska pracy (Linux)
7. Przygotowanie środowiska pracy (MacOS)
8. Przygotowanie środowiska pracy (Windows)
9. "Hello World" - pierwszy program w Pythonie

▼ Moduł 2 - Dogadać się z Pythonem

1. Interaktywna konsola
2. Zmienne
3. Zmienne (zadania)
4. Zmienne (rozwiązania)
5. Funkcja print i napisy
6. Funkcja print i napisy (zadania)
7. Funkcja print i napisy (rozwiązania)
8. Funkcja input
9. Funkcja input (zadania)

10. Funkcja input (rozwiązania)
11. Komentarze w kodzie
12. Podsumowanie
13. Quiz po module

▼ **Moduł 3 - Podstawowe typy zmiennych**

1. Wprowadzenie
2. Co wynika z różnicy typów?
3. Typ int oraz float
4. Typ int oraz float (zadanie)
5. Typ int oraz float (rozwiązanie)
6. Typ str
7. Typ str, czyli napisy (zadania)
8. Typ str, czyli napisy (rozwiązania)
9. Typ list I
10. Typ list II
11. Typ list (zadania)
12. Typ list (rozwiązania)
13. Typ dict, czyli słownik
14. Typ dict, czyli słownik (zadania)
15. Typ dict, czyli słownik (rozwiązania)
16. Typ NoneType
17. Podsumowanie
18. Quiz po module

▼ **Moduł 4 - Instrukcje warunkowe i operatory logiczne**

1. Wprowadzenie
2. Porównanie i typ bool

3. Porównanie i typ bool (zadania)
4. Porównanie i typ bool (rozwiązania)
5. Instrukcja if/else
6. Instrukcja if/else (zadania)
7. Instrukcja if/else (rozwiązania)
8. Operatory logiczne and/or/not
9. Operatory logiczne and/or/not (zadania)
10. Operatory logiczne and/or/not (rozwiązania)
11. Instrukcja elif
12. Instrukcja elif (zadania)
13. Instrukcja elif (rozwiązania)
14. Operatory in oraz is
15. Operatory in oraz is (zadania)
16. Operatory in oraz is (rozwiązania)
17. Podsumowanie
18. Quiz po module

▼ Moduł 5 - Pętle

1. Wprowadzenie
2. Pętla while
3. Pętla while (zadania)
4. Pętla while (rozwiązania)
5. Pętla for
6. Pętla for (zadania)
7. Pętla for (rozwiązania)
8. Pętla for in range
9. Pętla for in range (zadania)
10. Pętla for in range (rozwiązania)

11. Komendy break i continue
12. Komendy break I continue (zadania)
13. Komendy break I continue (rozwiązania)
14. Podsumowanie
15. Quiz po module

▼ Moduł 6 - Funkcje

1. Funkcje I
2. Funkcje II
3. Funkcje (zadania)
4. Funkcje (rozwiązania)
5. Argumenty pozycyjne i nazwane
6. Argumenty pozycyjne i nazwane (zadania)
7. Argumenty pozycyjne i nazwane (rozwiązania)
8. Zakres zmiennych
9. Parametry domyślne
10. Parametry domyślne (zadania)
11. Parametry domyślne (rozwiązania)
12. Podsumowanie
13. Quiz po module

▼ Moduł 7 - Podsumowanie kursu "Zrozumieć Programowanie"

1. Wstęp do podsumowania
2. Czym właściwie jest programowanie
3. Co dalej?

PythON: Początek

▼ Moduł 1 - Niezbędnik Adepta

1. Przywitanie w kursie "PythON: Początek"
2. Programistyczna droga i plan kursu
3. Czym jest programowanie?
4. Język i interpreter Python oraz PyCharm
5. Instalacja Pythona i PyCharma
6. Instalacja Pythona i PyCharma (Windows)
7. Instalacja Pythona i PyCharma (Linux)
8. Instalacja Pythona i PyCharma (MacOS)
9. Różne wersje Pythona
10. Alternatywy dla PyCharma
11. Hello World!
12. Python - filozofia
13. Ekspresowe przypomnienie
14. Debugowanie w PyCharmie
15. Pakiety i moduły
16. Pakiety i moduły (zadania)
17. Pakiety i moduły (rozwiązania)
18. Różne sposoby importowania
19. Różne sposoby importowania (zadania)
20. Różne sposoby importowania (rozwiązania)
21. Git - wprowadzenie
22. Czym jest system kontroli wersji?
23. Czym jest Git?
24. Git vs GitHub
25. Instalacja Git
26. Instalacja Git (Windows)

27. Instalacja Git (Linux)
28. Instalacja Git (MacOS)
29. Interfejs Gita i konfiguracja
30. Utworzenie repozytorium i pierwszy commit
31. Pobieranie istniejącego repozytorium
32. Git - historia zmian
33. Podsumowanie
34. Quiz po module

▼ **Moduł 2 - Programowanie obiektowe - część 1**

1. Powitanie
2. Paradygmat programowania
3. Programowanie proceduralne i obiektowe
4. Klasy i obiekty
5. Klasy i obiekty (zadania)
6. Klasy i obiekty (rozwiązania)
7. Konstruktor i pola obiektu
8. Konstruktor i pola obiektu (zadania)
9. Konstruktor i pola obiektu (rozwiązania)
10. Metoda, czyli zachowanie obiektu
11. Metoda, czyli zachowanie obiektu (zadania)
12. Metoda, czyli zachowanie obiektu (rozwiązania)
13. Czym są magiczne metody?
14. Metoda `__str__`
15. Metoda `__repr__`
16. Metody `__in__` oraz `__float__`
17. Metoda `__len__`
18. Metoda `__bool__`

19. Metody `__add__`, `__sub__`, `__mul__` i `__truediv__`
20. Metoda `__eq__` i inne metody porówniania
21. Metody magiczne (zadania)
22. Metody magiczne (rozwiązania)
23. Czym jest enkapsulacja?
24. Enkapsulacja w Pythonie
25. Enkapsulacja w Pythonie (zadania)
26. Enkapsulacja w Pythonie (rozwiązania)
27. Metody i pola klasy
28. Metody statyczne
29. Metody i pola klasy oraz metody statyczne (zadania)
30. Metody i pola klasy oraz metody statyczne (rozwiązania)
31. Funkcja jako obiekt
32. Funkcja jako obiekt (zadania)
33. Funkcja jako obiekt (rozwiązania)
34. Lambda - funkcja anonimowa
35. Lambda - funkcja anonimowa (zadania)
36. Lambda - funkcja anonimowa (rozwiązania)
37. Unpacking operators
38. Unpacking operators (zadania)
39. Unpacking operators (rozwiązania)
40. Podsumowanie
41. Quiz po module

▼ Moduł 3 - Programowanie obiektowe - część 2

1. Wprowadzenie
2. Prosta refaktoryzacja
3. Prosta refaktoryzacja (zadania)

4. Prosta refaktoryzacja (rozwiązania)
5. Property jako getter
6. Property jako getter dla wartości dynamicznej
7. Property jako setter
8. Property (zadania)
9. Property (rozwiązania)
10. Czym jest dziedziczenie?
11. Rozszerzenie typu bazowego
12. Dostęp do pól i metod z poziomu dziecka i rodzica
13. Dziedziczenie (zadania)
14. Dziedziczenie (rozwiązania)
15. Napisanie metody w klasie potomnej
16. Napisanie metody w klasie potomnej (zadania)
17. Napisanie metody w klasie potomnej (rozwiązania)
18. Dziedziczenie po typach wbudowanych
19. Abstrakcja i polimorfizm
20. Abstrakcja i polimorfizm (zadania)
21. Abstrakcja i polimorfizm (rozwiązania)
22. Metoda super
23. Metoda super (zadania)
24. Metoda super (rozwiązania)
25. Wielodziedziczenie i mixiny
26. MRO
27. Obiektość - dobre praktyki w Pythonie
28. Dobre praktyki - reguły SOLID, DRY
29. Podsumowanie
30. Quiz po module

▼ Moduł 4 - Wbudowane struktury danych

1. Powitanie
2. int oraz float
3. int oraz float (zadania)
4. int oraz float (rozwiązania)
5. str, czyli napisy
6. str, czyli napisy (zadania)
7. str, czyli napisy (rozwiązania)
8. Lista oraz list comprehensions
9. Lista oraz list comprehensions (zadania)
10. Lista oraz list comprehensions (rozwiązania)
11. Słownik oraz dict comprehensions
12. Słownik oraz dict comprehensions (zadania)
13. Słownik oraz dict comprehensions (rozwiązania)
14. Set, czyli zbiór
15. Set, czyli zbiór (zadania)
16. Set, czyli zbiór (rozwiązania)
17. Mutable oraz immutable
18. Frozenset
19. Frozenset (zadania)
20. Frozenset (rozwiązania)
21. Tuple
22. Tuple (zadania)
23. Tuple (rozwiązania)
24. Named tuple
25. Named tuple (zadania)
26. Named tuple (rozwiązania)

27. Dataclass
28. Dataclass (zadania)
29. Dataclass (rozwiązania)
30. Enum
31. Enum (zadania)
32. Enum (rozwiązania)
33. Podsumowanie
34. Quiz po module

▼ **Moduł 5 - Środowisko i biblioteki**

1. Wprowadzenie
2. Czym są biblioteki?
3. Virtualenv
4. Instalacja bibliotek
5. Przykład dla Windowsa
6. Linux - instalacja programu pip
7. Środowisko wirtualne z poziomu konsoli
8. Skąd biorą się biblioteki i jak je znaleźć?
9. Proste zarządzanie zależnościami
10. Zależności developerskie i produkcyjne
11. Virtualenvwrapper
12. Instalacja i pierwsze uruchomienie (Linux & MacOS)
13. Instalacja i pierwsze uruchomienie (Windows)
14. Virtualenvwrapper - przydatne komendy
15. Pipenv
16. Przydatne biblioteki
17. Przydatne biblioteki - Black
18. Przydatne biblioteki - GeoPy

19. Przydatne biblioteki - Folium
20. Przydatne biblioteki - openpyxl
21. Przydatne biblioteki - pydantic
22. Przydatne biblioteki - requests i inne
23. Podsumowanie
24. Quiz po module

▼ Moduł 6 - Obsługa błędów i pliki

1. Powitanie
2. Wyjątki w Pythonie
3. Wyjątki w Pythonie (zadania)
4. Wyjątki w Pythonie (rozwiązania)
5. Tworzenie własnych wyjątków
6. Tworzenie własnych wyjątków (zadania)
7. Tworzenie własnych wyjątków (rozwiązania)
8. Blok try... except..., czyli łapanie wyjątków
9. Blok try... except..., czyli łapanie wyjątków (zadania)
10. Blok try... except..., czyli łapanie wyjątków (rozwiązania)
11. Bloki finally oraz else
12. Bloki finally oraz else (zadania)
13. Bloki finally oraz else (rozwiązania)
14. Praca z plikami
15. Praca z plikami - konstrukcja with
16. Praca z plikami - konstrukcja with (zadania)
17. Praca z plikami - konstrukcja with (rozwiązania)
18. Praca z plikami - format CSV
19. Praca z plikami - format CSV (zadania)

20. Praca z plikami - format CSV (rozwiązania)
21. Praca z plikami - format JSON
22. Praca z plikami - format JSON (zadania)
23. Praca z plikami - format JSON (rozwiązania)
24. Podsumowanie
25. Quiz po module

▼ Moduł 7 - Scraping [BONUS]

1. Powitanie
2. Czym jest web scraping?
3. Jak zbudowana jest strona - HTML
4. Jak zbudowana jest strona - HTML (zadania)
5. Protokół HTTP
6. Programistyczny klient HTTP - biblioteka requests
7. Programistyczny klient HTTP - biblioteka requests (zadania)
8. Programistyczny klient HTTP - biblioteka requests (rozwiązania)
9. Proste przetwarzanie treści - operacje na str
10. Proste przetwarzanie treści - operacje na str (zadania)
11. Proste przetwarzanie treści - operacje na str (rozwiązania)
12. Biblioteka BeautifulSoup
13. Biblioteka BeautifulSoup (zadania)
14. Biblioteka BeautifulSoup (rozwiązania)
15. Podsumowanie
16. Do zobaczenia

▼ Moduł 8 - Certyfikacja Python Institute (przygotowanie do certyfikatu PECP-30-02)

1. Informacje przed przystąpieniem do certyfikacji
2. Wprowadzenie
3. Typ int - różne notacje
4. Typ int - różne notacje (zadania)
5. Typ int - różne notacje (rozwiązanie)
6. Operator floor division
7. Priorytet operatorów
8. Python keywords
9. Instrukcja else dla pętli
10. System dwójkowy i operatory bitowe
11. Platforma edube.org

▼ Moduł 9 - Test końcowy

1. Test końcowy

PythON: Wtajemniczenie

▼ Moduł 1 - Niezbędnik Adepta

1. Przywitanie
2. Programistyczna droga i plan kursu
3. Instalacja Pythona i PyCharm
4. Instalacja Pythona i PyCharm (Windows)
5. Instalacja Pythona i PyCharm (Linux)
6. Instalacja Pythona i PyCharm (MacOS)
7. Instalacja Gita
8. Instalacja Gita (Windows)

9. Instalacja Gita (Linux)
10. Instalacja Gita (MacOS)
11. Ekspresowe przypomnienie - środowisko i biblioteki
12. Ekspresowe przypomnienie - Git
13. Ekspresowe przypomnienie - programowanie obiektowe
14. Ekspresowe przypomnienie - typy danych
15. Ekspresowe przypomnienie - obsługa błędów
16. Ekspresowe przypomnienie - praca z plikami
17. Podsumowanie

▼ Moduł 2 - Funkcje

1. Wprowadzenie
2. Argumenty tylko nazwane
3. Argumenty tylko nazwane (zadanie)
4. Argumenty tylko nazwane (zadanie - wprowadzenie)
5. Argumenty tylko nazwane (rozwiązanie)
6. Argumenty tylko pozycyjne
7. Argumenty - kontekst i zastosowania
8. Argumenty tylko pozycyjne i nazwane (zadanie)
9. Argumenty tylko pozycyjne i nazwane (rozwiązanie)
10. Inner functions
11. Inner functions (zadania)
12. Inner functions (rozwiązania)
13. Rekurencja
14. Rekurencja (zadanie)
15. Rekurencja (rozwiązanie)
16. Podsumowanie

▼ Moduł 3 - Data i czas

1. Wprowadzenie
2. Data i czas - koncepcje
3. Date
4. Date (zadanie)
5. Date (rozwiązanie)
6. Data w formacie ISO
7. Data w formacie ISO (zadanie)
8. Data w formacie ISO (rozwiązanie)
9. Aktualna data
10. Aktualna data (zadanie)
11. Aktualna data (rozwiązanie)
12. UNIX Timestamp
13. Inne sposoby na utworzenie daty
14. Dane o dacie - metody obiektu date
15. Metody obiektu date (zadanie)
16. Metody obiektu date (rozwiązanie)
17. Porównywanie dat
18. Porównywanie dat (zadanie)
19. Porównywanie dat (rozwiązanie)
20. Typ time
21. Typ time (zadanie)
22. Typ time (rozwiązanie)
23. Czas w formacie ISO
24. Porównywanie czasu
25. Porównywanie czasu (zadanie)
26. Porównywanie czasu (rozwiązanie)
27. Podsumowanie

▼ Moduł 4 - Datetime, interwały czasowe i formatowanie

1. Wprowadzenie
2. Typ datetime
3. Typ datetime (zadanie)
4. Typ datetime (rozwiązanie)
5. Aktualna data i czas
6. Aktualna data i czas (zadanie)
7. Aktualna data i czas (rozwiązanie)
8. Datetime a date i time
9. Datetime a date i time (zadanie)
10. Datetime a date i time (rozwiązanie)
11. Datetime w formacie ISO
12. Datetime w formacie ISO (zadanie)
13. Datetime w formacie ISO (rozwiązanie)
14. Data i czas w różnych postaciach
15. Data i czas w różnych postaciach (zadanie)
16. Data i czas w różnych postaciach (rozwiązanie)
17. Formatowanie daty i czasu
18. Formatowanie daty i czasu (zadanie)
19. Formatowanie daty i czasu (rozwiązanie)
20. Parsowanie daty i czasu z różnych formatów
21. Parsowanie daty i czasu z różnych formatów (zadanie)
22. Parsowanie daty i czasu z różnych formatów (rozwiązanie)
23. Porównywanie obiektów datetime
24. Timedelta
25. Timedelta (zadanie)
26. Timedelta (rozwiązanie)

27. Podsumowanie

▼ Moduł 5 - Strefy czasowe

1. Wprowadzenie
2. Strefy czasowe
3. Obiekty aware - today, now, fromtimestamp
4. Obiekty aware - wczytywanie z ISO format
5. Aware datetime a date oraz time
6. Podstawienie strefy czasowej i konwersja
7. Niebezpieczne metody: utcnow i utcfromtimestamp
8. Strefy czasowe (zadanie)
9. Strefy czasowe (rozwiązania)
10. Strefy czasowe przed Pythonem 3.9
11. W jaki sposób przechowywać informacje o czasie
12. Podsumowanie

▼ Moduł 6 - Podstawy wyrażeń regularnych

1. Wprowadzenie
2. Metaznaki - klasa znaków
3. Metaznaki - klasa znaków (zadanie)
4. Metaznaki - klasa znaków (rozwiązanie)
5. Metaznaki - specjalne symbole
6. Metaznaki - specjalne symbole (zadania)
7. Metaznaki - specjalne symbole (rozwiązania)
8. Raw strings
9. Określanie liczby wystąpień
10. Określanie liczby wystąpień (zadanie)
11. Określanie liczby wystąpień (rozwiązanie)
12. Grupowanie

13. Grupowanie (zadanie)
14. Grupowanie (rozwiązanie)
15. Metody match i fullmatch
16. Metody match i fullmatch (zadanie)
17. Metody match i fullmatch (rozwiązanie)
18. Metoda search
19. Metody findall i finditer
20. Metody findall i finditer (zadanie)
21. Metody findall i finditer (rozwiązanie)
22. Flagi wyrażeń regularnych
23. Flagi wyrażeń regularnych (zadanie)
24. Flagi wyrażeń regularnych (rozwiązanie)
25. Kompilacja a wydajność
26. Podsumowanie

▼ Moduł 7 - Type hint

1. Wprowadzenie
2. Typowanie dynamiczne i statyczne
3. Po co nam statyczna kontrola typów?
4. Type hints w funkcjach
5. Type hints w funkcjach (zadanie)
6. Type hints w funkcjach (rozwiązanie)
7. Kontrola zgodności typów - mypy
8. Kontrola zgodności typów - mypy (zadanie)
9. Kontrola zgodności typów - mypy (rozwiązanie)
10. Typowanie zmiennych i wnioskowanie typów
11. Typowanie zmiennych i wnioskowanie typów (zadanie)
12. Typowanie zmiennych i wnioskowanie typów (rozwiązanie)

13. Deklarowanie złożonych typów
14. Deklarowanie złożonych typów (zadanie)
15. Deklarowanie złożonych typów (rozwiązanie)
16. Generyki
17. Przypadki szczególne - Any
18. Rzutowanie typu
19. Przypadki szczególne - ignore
20. Podsumowanie

▼ **Moduł 8 - Abstrakcja, interfejs i Protocol**

1. Wprowadzenie
2. Czym jest interfejs i API?
3. Klasy abstrakcyjne w Pythonie
4. Klasy abstrakcyjne w Pythonie (zadanie)
5. Klasy abstrakcyjne w Pythonie (rozwiązanie)
6. Protocol, czyli Structural subtyping
7. Protocol, czyli Structural subtyping (zadanie)
8. Protocol, czyli Structural subtyping (rozwiązanie)
9. Typ jako część interfejsu
10. Stub files
11. Stub files (zadanie)
12. Stub files (rozwiązanie)
13. Podsumowanie

▼ **Moduł 9 - Iterator i generator**

1. Wprowadzenie
2. Protokół iteratora
3. Protokół iteratora (zadanie)
4. Protokół iteratora (rozwiązanie)

5. Generator
6. Generator (zadanie)
7. Generator (rozwiązanie)
8. Metoda send
9. Metody throw i close
10. Podsumowanie

▼ **Moduł 10 - Context Manager**

1. Wprowadzenie
2. Własny Context Manager
3. Własny Context Manager (zadania)
4. Własny Context Manager (rozwiązania)
5. Funkcja generator i context manager
6. Funkcja generator i context manager (zadania)
7. Funkcja generator i context manager (rozwiązania)
8. Reużywalne context managery
9. Podsumowanie

▼ **Moduł 11 - Moduł Collections - część 1**

1. Wprowadzenie
2. defaultdict
3. defaultdict (zadania)
4. defaultdict (rozwiązania)
5. OrderedDict
6. OrderedDict (zadania)
7. OrderedDict (rozwiązania)
8. deque
9. deque (zadania)
10. deque (rozwiązania)

11. Podsumowanie

▼ **Moduł 12 - Moduł Collections - część 2**

1. Wprowadzenie
2. Counter
3. Counter (zadania)
4. Counter (rozwiązania)
5. ChainMap
6. ChainMap (zadania)
7. ChainMap (rozwiązania)
8. Abstract Base Classes
9. Podsumowanie

▼ **Moduł 13 - Dekoratory**

1. Wprowadzenie
2. Dekorator
3. Dekorator (zadanie)
4. Dekorator (rozwiązanie)
5. Dekorowanie funkcji z argumentami
6. Dekorowanie funkcji z argumentami (zadanie)
7. Dekorowanie funkcji z argumentami (rozwiązanie)
8. Dekorator z argumentami
9. Dekorator z argumentami (zadanie)
10. Dekorator z argumentami (rozwiązanie)
11. `functools.wrap`
12. Dekorowanie klasy
13. Dekorator `total_ordering`
14. Dekorator `total_ordering` (zadanie)
15. Dekorator `total_ordering` (rozwiązanie)

16. Dekorator lru_cache
17. Dekorator lru_cache (zadanie)
18. Dekorator lru_cache (rozwiązanie)
19. Podsumowanie

▼ Moduł 14 - Metaklasy

1. Wprowadzenie
2. Jak Python tworzy klasy?
3. Dynamiczne definiowanie klasy (zadanie)
4. Dynamiczne definiowanie klasy (rozwiązanie)
5. Weryfikacja poprawności klasy
6. Weryfikacja poprawności klasy (zadanie)
7. Weryfikacja poprawności klasy (rozwiązanie)
8. Rejestracja klas
9. Rejestracja klas (zadanie)
10. Rejestracja klas (rozwiązanie)
11. Modyfikacja definicji klasy w locie
12. Podsumowanie

▼ Moduł 15 - Wybrane wzorce projektowe

1. Wprowadzenie
2. Factory method
3. Factory method (zadanie)
4. Factory method (rozwiązanie)
5. Abstract factory
6. Abstract factory (zadanie)
7. Abstract factory (rozwiązanie)
8. Builder
9. Builder (zadanie)

10. Builder (rozwiązanie)
11. Adapter
12. Adapter (zadanie)
13. Adapter (rozwiązanie)
14. Fasada
15. Fasada (zadanie)
16. Fasada (rozwiązanie)
17. Obserwator
18. Dekorator
19. Iterator
20. Podsumowanie

▼ Moduł 16 - Testy - część 1

1. Wprowadzenie
2. Biblioteka pytest i proste testy
3. Testy (zadanie)
4. Testy (rozwiązanie)
5. Testowanie wyjątków
6. Testowanie wyjątków (zadanie)
7. Testowanie wyjątków (rozwiązanie)
8. Fixtury
9. Fixtury (zadanie)
10. Fixtury (rozwiązanie)
11. Podsumowanie

▼ Moduł 17 - Testy - część 2

1. Wprowadzenie
2. Czym są stuby i mocki?
3. Stuby i mocki (zadanie)

4. Stuby i mocki (rozwiązanie)
5. Moduł unittest
6. Moduł unittest (zadanie)
7. Moduł unittest (rozwiązanie)
8. pytest a unittest
9. Test Driven Development
10. Podsumowanie

▼ Moduł 18 - Współbieżność - część 1

1. Wprowadzenie
2. Czym jest współbieżność?
3. Współbieżność i równoległość
4. Współbieżność i równoległość (zadanie)
5. Wątki w Pythonie
6. Wątki w Pythonie (zadanie)
7. Wątki w Pythonie (rozwiązanie)
8. Model z wyłączeniem
9. Wyścigi, sekcja krytyczna i synchronizacja
10. Wyścigi i synchronizacja (zadanie)
11. Wyścigi i synchronizacja (rozwiązanie)
12. Podsumowanie

▼ Moduł 19 - Współbieżność - część 2

1. Wprowadzenie
2. Wątki i moduł queue
3. Wątki i moduł queue (zadanie)
4. Wątki i moduł queue (rozwiązanie)
5. CPU bounded vs IO bounded
6. Global Interpreter Lock

7. Procesy
8. Procesy (zadanie)
9. Procesy (rozwiązanie)
10. Skalowalność
11. Komunikacja międzyprocesowa
12. Wątki i procesy razem
13. Subprocess
14. Podsumowanie

▼ **Moduł 20 - Współbieżność - część 3**

1. Wprowadzenie
2. Wykonanie asynchroniczne i model kooperatywny
3. Wykonanie asynchroniczne i model kooperatywny (zadanie)
4. Wykonanie asynchroniczne i model kooperatywny (rozwiązanie)
5. Podsumowanie

▼ **Moduł 21 - Warsztat programistyczny: Git i workflow**

1. Wprowadzenie
2. Git - branche
3. Git - branche (zadanie)
4. Git - mergowanie
5. Git - rozwiązywanie konfliktów
6. Git - tagi
7. Pull requests i code review
8. Podsumowanie

▼ **Moduł 22 - Warsztat programistyczny: workflow i CI**

1. Wprowadzenie
2. Gitflow
3. Githubflow
4. Proces CI - wprowadzenie
5. Proces CI - testy automatyczne
6. Proces CI - kontrola typów i linter
7. Proces CI - formatowanie
8. Podsumowanie

▼ **Moduł 23 - Podsumowanie i test końcowy**

1. Podsumowanie kursu i kilka wskazówek
2. Test końcowy