

kod szkolenia: PYTH_LUW / PL AA 5d

Python dla administratorów systemów LUW

Szkolenie: Python dla administratorów systemów LUW



Przeznaczenie szkolenia

Szkolenie skierowane jest do osób chcących nauczyć się tworzenia skryptów do automatyzacji zadań na urządzeniach opartych o **środowiska LUW** przy użyciu języka **Python**.



Korzyści wynikające z ukończenia szkolenia

Biegłość w tworzeniu skryptów w języku Python pozwala na automatyzację żmudnych zadań administracyjnych zwiększając efektywność i jakość pracy administratorów systemów. **Język Python** zapewnia bogaty zestaw narzędzi do integracji z technologiami występującymi w **środowiskach Linux i Windows**.



Oczekiwane przygotowanie słuchaczy

- Znajomość środowiska Linux/Windows
- Umiejętność analitycznego myślenia.

Rekomendowana bibliografia:

"Python for Unix and Linux System Administrator" – Autor: Noab Gift & Jeremy M. Jones



Język szkolenia

- Szkolenie: polski



Szkolenie obejmuje

Metoda szkolenia:

- Warsztat + wykład



Czas trwania

5 dni / 35 godzin

Agenda szkolenia

1. Wstęp

- Cel szkolenia
- Charakterystyka języka Python
- Środowisko programistyczne – PyCharm, VisualStudio Code czy inne?
- Uruchamianie programów w Python
 - w środowisku Windows
 - w środowisku Linux

2. Podstawy języka Python

- Podstawowe właściwości języka
- Składnia
- Podstawowe typy danych
- Struktury danych
- Instrukcje warunkowe i pętle
- Funkcje
- Obiekty
- Obsługa błędów i asercje
- Moduły, pakiety i importowanie
- Biblioteki podstawowe
- Elementy programowania funkcyjnego

3. Przetwarzanie tekstu

- Podstawowe operacje na tekstach
- Wczytywanie i parsowanie plików tekstowych – context managers

- Wyrażenia regularne
- Strumieniowe przetwarzanie plików tekstowych
- 4. System plików i dane
 - Interakcja z systemem plików:
 - Parsowanie drzewa katalogów
 - Podstawowe operacje na plikach
 - Prawa dostępu
 - Przetwarzanie ścieżek systemowych
 - Interakcja z danymi – przegląd formatów i dostępnych bibliotek
 - Serializacja danych – formaty pickle, JSON, XML i inne
 - Archiwizacja i kompresowanie
- 5. Przetwarzanie współbieżne i rozproszona dystrybucja zadań
 - Wykonywanie innych programów z Pythona
 - Programy wielowątkowe i wieloprocesowe – przegląd problemów i rozwiązań
 - Dystrybucja zadań na wiele procesów – multiprocessing, concurrent.futures i celery
- 6. Sieci
 - FTP / SFTP / SCP
 - Email – konstrukcja wiadomości i protokoły SMTP, POP3 i IMAP
 - Protokół http i serwisy REST
 - SSH i telnet – paramiko
 - Komunikacja z urządzeniami sieciowymi – netmiko
 - Zdalne zarządzanie serwerami – ansible
- 7. Python – dobre praktyki
 - Wirtualizacja środowiska
 - Type hinting i docstrings
 - Programy python jako skrypty systemowe
 - Parsowanie parametrów wywołania
 - Przenośność kodu
 - Wybór najlepszej biblioteki do integracji z wybraną technologią