



Notebook Master
Sp. z o.o.

★★★★★ 4,8 / 5

188 ocen

Python / Etap I / Podstawy programowania w języku Python - szkolenie

Numer usługi 2025/08/20/158529/2948849

📍 zdalna w czasie rzeczywistym

🏠 Usługa szkoleniowa

🕒 40 h

📅 17.11.2025 do 21.11.2025

3 567,00 PLN brutto

2 900,00 PLN netto

89,18 PLN brutto/h

72,50 PLN netto/h

Informacje podstawowe

Kategoria

Informatyka i telekomunikacja / Programowanie

Identyfikatory projektów

Małopolski Pociąg do kariery, Zachodniopomorskie Bony Szkoleniowe,
Kierunek - Rozwój, Regionalny Fundusz Szkoleniowy II, FELB.06.08-IZ.00-
0083/24

Grupa docelowa usługi

Szkolenie skierowane jest zarówno do osób fizycznych, jak i do przedsiębiorców i ich pracowników pracujących w branży IT, chcących nabyć umiejętności programowania w języku Python i wykorzystać ją w ramach prowadzonej działalności gospodarczej i etatu.

Usługa rozwojowa adresowa również dla Uczestników projektów, m.in.:

- Małopolski pociąg do kariery
- Zachodniopomorskie Bony Szkoleniowe
- Kierunek – Rozwój
- Regionalny Fundusz Szkoleniowy II
- Lubuskie Bony Rozwojowe
- Usługi rozwojowe dla mieszkańców województwa lubuskiego

Minimalna liczba uczestników

1

Maksymalna liczba uczestników

8

Data zakończenia rekrutacji

16-11-2025

Forma prowadzenia usługi

zdalna w czasie rzeczywistym

Liczba godzin usługi

40

Podstawa uzyskania wpisu do BUR

Znak Jakości Małopolskich Standardów Usług Edukacyjno-Szkoleniowych (MSUES) - wersja 2.0

Cel

Cel edukacyjny

Usługa "Python / Etap I / Podstawy programowania w języku Python.", prowadzi do nabycia specjalistycznych kompetencji w obszarze tematycznym szkolenia i przygotowuje do samodzielnego i prawidłowego wykonywania obowiązków w zakresie podstaw programowania z wykorzystaniem języka Python.

Efekty uczenia się oraz kryteria weryfikacji ich osiągnięcia i Metody walidacji

Efekty uczenia się	Kryteria weryfikacji	Metoda walidacji
Pobiera i instaluje Python.	Rozpoznaje standardową implementację języka Python.	Test teoretyczny
	Wskazuje poprawną komendę do sprawdzenia wersji Pythona w terminalu.	Test teoretyczny
Sprawdza wersję Pythona.	Wybiera polecenie zwracające wersję interpretera Python.	Test teoretyczny
Instaluje edytor kodu, np. Visual Studio Code.	Dobiera edytor kodu wspierający Python.	Test teoretyczny
	Porównuje instalację user install i system install.	Test teoretyczny
Rozróżnia typy danych, operatory arytmetyczne, porównania, przypisania (w tym skrótowych) i logiczne.	Identyfikuje typy danych: int, float, complex, str, bool.	Test teoretyczny
	Rozpoznaje działanie operatorów arytmetycznych, porównań, przypisania (np. +=) i logicznych.	Test teoretyczny
Korzysta z terminala do uruchamiania skryptów Pythona i komunikacji z terminalem i zarządzania zmiennymi środowiskowymi.	Wybiera poprawne polecenie do uruchomienia skryptu Pythona z terminala.	Test teoretyczny
	Wskazuje zastosowanie zmiennych środowiskowych w kontekście Pythona.	Test teoretyczny
Pracuje z operacjami na plikach, takimi jak otwieranie, zapisywanie i odczytywanie, oraz obsługą błędów związanymi z operacjami na plikach.	Rozpoznaje tryby otwierania plików (r, w, a).	Test teoretyczny
	Wybiera fragment kodu obsługujący błędy przy operacjach na plikach.	Test teoretyczny

Efekty uczenia się	Kryteria weryfikacji	Metoda walidacji
Tworzy interfejs graficzny, w tym okna, widżety oraz układ i rozmieszczenie elementów interfejsu.	Wybiera widżet odpowiadający za etykietę, pole tekstowe i przycisk w tkinter.	Test teoretyczny
	Rozpoznaje metody rozmieszczania elementów (pack, grid, place).	Test teoretyczny
Korzysta z Jupyter Notebook do tworzenia komórek kodu.	Rozróżnia komórki kodu i komórki tekstowe (Markdown).	Test teoretyczny
	Rozpoznaje sposób uruchamiania komórek w notatniku Jupyter.	Test teoretyczny
Wprowadza pojęcie enkapsulacji, praca z lokalnymi modułami. Instaluje dodatkowe moduły i importuje je.	Charakteryzuje enkapsulację w Pythonie.	Test teoretyczny
	Rozpoznaje poprawną składnię importu modułu lokalnego.	Test teoretyczny
	Wybiera komendę pip install do instalacji pakietu.	Test teoretyczny
	Rozpoznaje poprawną instrukcję importu modułu w Pythonie.	Test teoretyczny
Tworzy i aktywuje środowisko wirtualne (venv). Instaluje pakiety w środowisku wirtualnym.	Wskazuje komendę tworzącą środowisko wirtualne	Test teoretyczny
	Rozpoznaje komendę aktywacji środowiska wirtualnego.	Test teoretyczny
	Stosuje polecenie pip install w aktywnym środowisku.	Test teoretyczny
	Potwierdza poprawność instalacji i dostępność pakietów w środowisku wirtualnym.	Test teoretyczny
Porównuje menedżery pakietów: Pip, Conda, Poetry.	Porównuje funkcjonalności pip, conda i poetry.	Test teoretyczny
	Wskazuje zalety i wady menedżera pakietów.	Test teoretyczny
Omawia typy danych liczbowych, takie jak int, float, complex oraz funkcje matematyczne z modułu 'math'.	Rozpoznaje typy danych liczbowych: int, float, complex.	Test teoretyczny
	Wskazuje poprawną funkcję matematyczną z modułu math.	Test teoretyczny

Efekty uczenia się	Kryteria weryfikacji	Metoda walidacji
Pracuje z napisami (string), w tym formatowaniem, indeksowaniem i wycinaniem fragmentów tekstu, oraz korzysta z metod dla napisów.	Rozpoznaje metody dla napisów (upper(), replace(), split()).	Test teoretyczny
	Określa sposób indeksowania i wycinania fragmentów napisów.	Test teoretyczny
Charakteryzuje kolekcje, takie jak listy, krotki, zbiory i słowniki, oraz wykonuje operacje na nich.	Rozpoznaje cechy i zastosowania list, krotek, zbiorów i słowników.	Test teoretyczny
	Wybiera poprawne operacje na kolekcjach (dodawanie, usuwanie, indeksowanie).	Test teoretyczny
Stosuje konstrukcje logiczne, takie jak operacje warunkowe (if, elif, else), pętle (for, while), list comprehension, set comprehension i dict comprehension. Definiuje funkcje za pomocą 'def' i korzysta z funkcji anonimowych (lambda).	Rozpoznaje poprawne użycie instrukcji warunkowych i pętli.	Test teoretyczny
	Wskazuje przykład użycia comprehension.	Test teoretyczny
	Wybiera poprawną składnię definicji funkcji.	Test teoretyczny
	Rozpoznaje zastosowanie funkcji lambda.	Test teoretyczny
Zwraca dane z funkcji za pomocą 'return' oraz obsługuje wartość 'None'. Pracuje z parametrami funkcji, argumentami pozycyjnymi, domyślnymi oraz korzysta z *args i **kwargs w funkcjach.	Określa zastosowanie return w celu zwrócenia wartości.	Test teoretyczny
	Rozpoznaje sytuacje, w których funkcja zwraca None.	Test teoretyczny
	Rozróżnia argumenty pozycyjne, domyślne, *args i **kwargs.	Test teoretyczny
	Wybiera poprawny przykład funkcji korzystającej z *args lub **kwargs.	Test teoretyczny

Kwalifikacje

Kompetencje

Usługa prowadzi do nabycia kompetencji.

Warunki uznania kompetencji

Pytanie 1. Czy dokument potwierdzający uzyskanie kompetencji zawiera opis efektów uczenia się?

TAK

Pytanie 2. Czy dokument potwierdza, że walidacja została przeprowadzona w oparciu o zdefiniowane w efektach uczenia się kryteria ich weryfikacji?

TAK

Pytanie 3. Czy dokument potwierdza zastosowanie rozwiązań zapewniających rozdzielenie procesów kształcenia i szkolenia od walidacji?

TAK

Program

Szkolenie skierowane jest zarówno do osób fizycznych, jak i do przedsiębiorców i ich pracowników pracujących w branży IT, chcących nabyć umiejętności programowania w języku Python i wykorzystać ją w ramach prowadzonej działalności gospodarczej i etatu.

Ramowy plan kształcenia:

I. Instalacja. (teoria+praktyka)

1. Pobieranie i instalacja interpretera Python.
2. Sprawdzanie wersji Pythona.
3. Semantyka, a prezentacja. Instalacja edytora kodu (na przykładzie Visual Studio Code).

II. Operatory i zmienne. (teoria+praktyka)

1. Typy danych - wprowadzenie.
2. Operatory arytmetyczne.
3. Operatory porównania.
4. Operatory przypisania.
5. Operatory logiczne.

III. Komunikacja z programem. (teoria+praktyka)

1. Terminal.
2. a: Uruchamianie skryptów Pythona.
3. b: Komunikacja z terminalem.
4. c: Zmienne środowiskowe.
5. Operacje na plikach.
6. a: Otwieranie, zapisywanie, odczytywanie.
7. b: Obsługa błędów.
8. Interfejs graficzny.
9. a: Tworzenie okien.
10. b: Tworzenie widgetów.
11. c: Układ i rozmieszczenie.
12. Jupyter Notebook.
13. a: Instalacja i uruchomienie.
14. b: Tworzenie komórek kodu i Markdown.

IV. Środowisko. (teoria+praktyka)

1. Enkapsulacja.
2. a: Wprowadzenie do koncepcji.
3. b: Lokalne moduły.
4. Instalacja dodatkowych modułów.
5. a: Importowanie modułów.
6. b: Użycie pip do instalacji pakietów.
7. Środowiska wirtualne.
8. a: Tworzenie i aktywacja środowiska wirtualnego (venv).
9. b: Instalacja pakietów w środowisku wirtualnym.
10. Pip/Anaconda/Poetry.
11. a: Różnice między menedżerami pakietów.
12. b: Instalacja i podstawowe komendy.

V. Typy danych - Szczegółowo. (teoria+praktyka)

1. Typy liczbowe.
2. a: Int, float, complex.
3. b: Funkcje matematyczne (math).
4. Tekst.
5. a: String.
6. b: Formatowanie, indeksowanie, wycinanie.
7. c: Metody stringów.
8. Kontenery.

9. a: Listy, krotki, zbiory, słowniki.

10. b: Operacje na kontenerach.

VI. Konstrukcje logiczne. (teoria+praktyka)

1. Operacje warunkowe i operator trójskładnikowy.

2. a: if, elif, else.

3. b: Warunki złożone.

4. c: Operator trójskładnikowy.

5. Pętle i wyrażenia kontenerów.

6. a: Pętle for, while.

7. b: Pętle zagnieżdżone.

8. c: List comprehension.

9. d: Set comprehension.

10. e: Dict comprehension.

VII. Funkcje. (teoria+praktyka)

1. Definicje def i lambda.

2. a: Tworzenie funkcji za pomocą def.

3. b: Funkcje anonimowe (lambda).

4. Zwracanie danych.

5. a: Instrukcja return.

6. b: Wartość None.

7. Parametry, args, kwargs.

8. a: Parametry pozycyjne, domyślne, nazwane.

9. b: *args i **kwargs.

10. Zmienne lokalne.

11. a: Zasięg zmiennych.

12. b: Zmienne globalne w funkcji.

13. c: Shadowing.

Szkolenie trwa 40 godzin dydaktycznych (przerwy nie wliczają się do czasu trwania usługi rozwojowej) i realizowane jest w kameralnych grupach, maksymalnie 8-osobowych. Każdy uczestnik realizujący szkolenie w formie zdalnej w czasie rzeczywistym ma możliwość otrzymania od nas (za pośrednictwem kuriera) wyposażenie stanowiska szkoleniowego (po ukończeniu szkolenia sprzęt zostaje odebrany przez kuriera).

Zawarto umowę z WUP w Szczecinie na świadczenie usług rozwojowych z wykorzystaniem elektronicznych bonów szkoleniowych.

Zawarto umowę z WUP w Toruniu w ramach Projektu Kierunek – Rozwój

Zaakceptowano Regulamin "Małopolskiego Pociągu do Kariery" dla instytucji szkoleniowych.

Wymagana jest frekwencja na poziomie 80%.

Obecność na zajęciach potwierdza raport z platformy Zoom.

Szkolenie prowadzone jest z wykorzystaniem metod nauczania aktywizujących uczestników: dyskusja w grupie, burza mózgów, ćwiczenia.

Harmonogram

Liczba przedmiotów/zajęć: 36

Przedmiot / temat zajęć	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
1 z 36 Pobieranie i instalacja interpretera Python. Sprawdzanie wersji Pythona. (Wykłady, dyskusja, ćwiczenia, testy.)	Karol Kołodziejczyk	17-11-2025	08:45	10:15	01:30
2 z 36 Przerwa.	Karol Kołodziejczyk	17-11-2025	10:15	10:30	00:15
3 z 36 Semantyka, a prezentacja. Instalacja edytora kodu (na przykładzie Visual Studio Code). (Wykłady, dyskusja, ćwiczeni)	Karol Kołodziejczyk	17-11-2025	10:30	12:00	01:30
4 z 36 Przerwa.	Karol Kołodziejczyk	17-11-2025	12:00	12:45	00:45
5 z 36 Operatory i zmienne. Typy danych - wprowadzenie. Operatory arytmetyczne. Operatory przypisania. Operatory logiczne. (Wykłady, dyskusja, ćwiczenia.)	Karol Kołodziejczyk	17-11-2025	12:45	14:15	01:30
6 z 36 Przerwa.	Karol Kołodziejczyk	17-11-2025	14:15	14:30	00:15

Przedmiot / temat zajęć	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
7 z 36 Komunikacja z programem. Terminal. Uruchamianie skryptów Pythona. Komunikacja z terminalem. Zmienne środowiskowe. (Wykłady, dyskusja, ćwiczenia.)	Karol Kołodziejczyk	17-11-2025	14:30	16:00	01:30
8 z 36 Operacje na plikach. Otwieranie, zapisywanie, odczytywanie. Obsługa błędów. (Wykłady, dyskusja, ćwiczenia.)	Karol Kołodziejczyk	18-11-2025	08:45	10:15	01:30
9 z 36 Przerwa.	Karol Kołodziejczyk	18-11-2025	10:15	10:30	00:15
10 z 36 Interfejs graficzny. Tworzenie okien. Tworzenie widgetów. Układ i rozmieszczenie. (Wykłady, dyskusja, ćwiczenia.)	Karol Kołodziejczyk	18-11-2025	10:30	12:00	01:30
11 z 36 Przerwa.	Karol Kołodziejczyk	18-11-2025	12:00	12:45	00:45
12 z 36 Jupyter Notebook. Instalacja i uruchomienie. Tworzenie komórek kodu i Markdown. (Wykłady, dyskusja, ćwiczenia.)	Karol Kołodziejczyk	18-11-2025	12:45	14:15	01:30
13 z 36 Przerwa.	Karol Kołodziejczyk	18-11-2025	14:15	14:30	00:15

Przedmiot / temat zajęć	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
14 z 36 Środowisko. Enkapsulacja. Wprowadzenie do koncepcji. Lokalne moduły. (Wykłady, dyskusja, ćwiczenia.)	Karol Kołodziejczyk	18-11-2025	14:30	16:00	01:30
15 z 36 Instalacja dodatkowych modułów. Importowanie modułów. Użycie pip do instalacji pakietów. (Wykłady, dyskusja, ćwiczenia.)	Karol Kołodziejczyk	19-11-2025	08:45	10:15	01:30
16 z 36 Przerwa.	Karol Kołodziejczyk	19-11-2025	10:15	10:30	00:15
17 z 36 Środowiska wirtualne. Tworzenie i aktywacja środowiska wirtualnego (venv). Instalacja pakietów w środowisku wirtualnym. (Wykłady, dyskusja, ćwiczenia.)	Karol Kołodziejczyk	19-11-2025	10:30	12:00	01:30
18 z 36 Przerwa.	Karol Kołodziejczyk	19-11-2025	12:00	12:45	00:45
19 z 36 Pip/Anaconda/Pytropy. Różnice między menedżerami pakietów. Instalacja i podstawowe komendy. (Wykłady, dyskusja, ćwiczenia.)	Karol Kołodziejczyk	19-11-2025	12:45	14:15	01:30

Przedmiot / temat zajęć	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
20 z 36 Przerwa.	Karol Kołodziejczyk	19-11-2025	14:15	14:30	00:15
21 z 36 Typy danych - Szczegółowo. Typy liczbowe. Int, float, complex. Funkcje matematyczne (math). (Wykłady, dyskusja, ćwiczenia.)	Karol Kołodziejczyk	19-11-2025	14:30	16:00	01:30
22 z 36 Tekst. String. Formatowanie, indeksowanie, wycinanie. Metody stringów. (Wykłady, dyskusja, ćwiczenia.)	Karol Kołodziejczyk	20-11-2025	08:45	10:15	01:30
23 z 36 Przerwa.	Karol Kołodziejczyk	20-11-2025	10:15	10:30	00:15
24 z 36 Kontenery. Listy, krotki, zbiory, słowniki. Operacje na kontenerach. (Wykłady, dyskusja, ćwiczenia.)	Karol Kołodziejczyk	20-11-2025	10:30	12:00	01:30
25 z 36 Przerwa.	Karol Kołodziejczyk	20-11-2025	12:00	12:45	00:45
26 z 36 Konstrukcje logiczne. Operacje warunkowe i operator trójskładnikowy. (Wykłady, dyskusja, ćwiczenia.)	Karol Kołodziejczyk	20-11-2025	12:45	14:15	01:30
27 z 36 Przerwa.	Karol Kołodziejczyk	20-11-2025	14:15	14:30	00:15

Przedmiot / temat zajęć	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
28 z 36 Pętle i wyrażenia kontenerów. Pętle for, while. Pętle zagnieżdżone. List/set/dict comprehension. (Wykłady, dyskusja, ćwiczenia.)	Karol Kołodziejczyk	20-11-2025	14:30	16:00	01:30
29 z 36 Funkcje. Definicje def i lambda. Tworzenie funkcji za pomocą def. Funkcje anonimowe (lambda). (Wykłady, dyskusja, ćwiczenia.)	Karol Kołodziejczyk	21-11-2025	08:45	10:15	01:30
30 z 36 Przerwa.	Karol Kołodziejczyk	21-11-2025	10:15	10:30	00:15
31 z 36 Zwracanie danych. Instrukcja return. Wartość None. (Wykłady, dyskusja, ćwiczenia.)	Karol Kołodziejczyk	21-11-2025	10:30	12:00	01:30
32 z 36 Przerwa.	Karol Kołodziejczyk	21-11-2025	12:00	12:45	00:45
33 z 36 Parametry, args, kwargs. Parametry pozycyjne, domyślne, nazwane. *args i **kwargs. (Wykłady, dyskusja, ćwiczenia.)	Karol Kołodziejczyk	21-11-2025	12:45	14:15	01:30
34 z 36 Przerwa.	Karol Kołodziejczyk	21-11-2025	14:15	14:30	00:15

Przedmiot / temat zajęć	Prowadzący	Data realizacji zajęć	Godzina rozpoczęcia	Godzina zakończenia	Liczba godzin
35 z 36 Zmienne lokalne. Zasięg zmiennych. Zmienne globalne w funkcji. Shadowing. (Wykłady, dyskusja, ćwiczenia, testy.)	Karol Kołodziejczyk	21-11-2025	14:30	15:30	01:00
36 z 36 Walidacja.	-	21-11-2025	15:30	16:00	00:30

Cennik

Cennik

Rodzaj ceny	Cena
Koszt przypadający na 1 uczestnika brutto	3 567,00 PLN
Koszt przypadający na 1 uczestnika netto	2 900,00 PLN
Koszt osobogodziny brutto	89,18 PLN
Koszt osobogodziny netto	72,50 PLN

Prowadzący

Liczba prowadzących: 1



1 z 1

Karol Kołodziejczyk

Obszar specjalizacji: Programowanie w językach JavaScript, Python i gdScript, tworzenie aplikacji i gier.

Doświadczenie zawodowe:

- 5 lat prowadzenia własnej działalności w branży informatycznej. Doświadczenie szkoleniowe:
- 3 lata pracy jako nauczyciel programowania w technikum i instruktor programowania na kursach zawodowych w TEB Edukacja.
- 3 lata doświadczenia szkoleniowego w zakresie HTML/CSS & JavaScript.
- 1 rok pracy jako nauczyciel programowania w Technikum Technologii Cyfrowych.

Wykształcenie: Techniczne - technik informatyk.

Informacje dodatkowe

Informacje o materiałach dla uczestników usługi

Całość opracowanych materiałów składa się z: opisów, ćwiczeń, poleceń, komend, procedur i zdjęć. Po zakończeniu kształcenia wszyscy uczestnicy otrzymują materiały w formie skryptu dotyczące całości przekazywanej wiedzy.

Sprzęt i materiały przesłane do uczestnika "zdalnego w czasie rzeczywistym" na czas trwania szkolenia (w celu realizacji części praktycznej szkolenia):

- jednostka sprzętowa z niezbędnym oprogramowaniem, peryferia

W/w wyposażenie stanowiska szkoleniowego jest przesłane i następnie odebrane od uczestnika "zdalnego w czasie rzeczywistym" po ukończeniu szkolenia za pośrednictwem kuriera.

Informacje dodatkowe

Faktura za usługę rozwojową podlega zwolnieniu z VAT dla osób korzystających z dofinansowania powyżej 70%.

Szkolenie trwa przez 5 dni co stanowi łącznie 40 godzin dydaktycznych (przerwy nie są wliczone do czasu trwania usługi).

Pierwsza przerwa zaczyna się 10:15 i kończy 10:30.

Druga przerwa zaczyna się 12:00 i kończy 12:45.

Trzecia przerwa zaczyna się 14:15 i kończy 14:30.

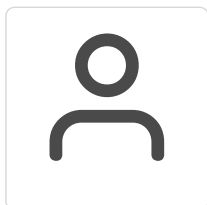
Szkolenie rozpoczyna się pre-testem weryfikującym początkową wiedzę uczestnika usługi rozwojowej i zakończone jest wewnętrznym egzaminem (post-test) weryfikującym i potwierdzającym pozyskaną wiedzę, pozytywne jego zaliczenie honorowane jest certyfikatem potwierdzającym jego ukończenie i uzyskane efekty kształcenia.

Warunki techniczne

Warunki techniczne niezbędne do udziału w usłudze:

- Do połączenia zdalnego w czasie rzeczywistym pomiędzy uczestnikami, a trenerem służy program "Zoom Client for Meetings" (do pobrania ze strony <https://zoom.us/download>).
- Komputer/laptop z kamerką internetową z zainstalowanym klientem Zoom, minimum dwurdzeniowy CPU o taktowaniu 2 GHz, min. 2 GB RAM.
- Mikrofon i słuchawki (ewentualnie głośniki).
- System operacyjny MacOS 10.7 lub nowszy, Windows 7, 8, 10, Linux: Mint, Fedora, Ubuntu, RedHat.
- Przeglądarkę internetową: Chrome 30 lub nowszy, Firefox 27 lub nowszy, Edge 12 lub nowszy, Safari 7 lub nowsze.
- Dostęp do internetu. Zalecane parametry przepustowości łączy: min. 5 Mbps - upload oraz min. 10 Mbps - download, zarezerwowane w danym momencie na pracę zdalną w czasie rzeczywistym. Umożliwi to komfortową komunikację pomiędzy uczestnikami, a trenerem. Link umożliwiający dostęp do szkolenia jest aktywny przez cały czas jego trwania, do końca zakończenia danego etapu szkolenia. Każdy uczestnik będzie mógł użyć go w dowolnym momencie trwania szkolenia.

Kontakt



Artur Kowalewski

E-mail szkolenia@notebookmaster.pl

Telefon (+48) 573 436 635