Dawid Kubik

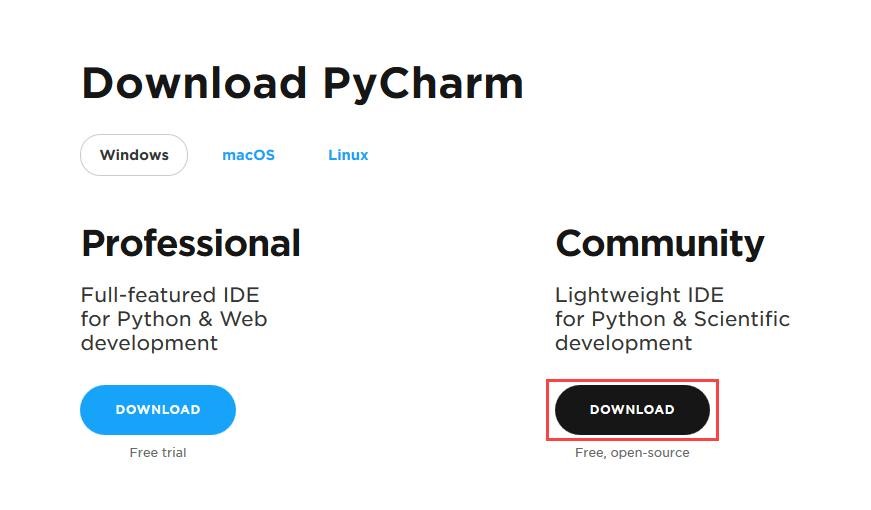
I Dzial instalowanie pycharm

Zad 2

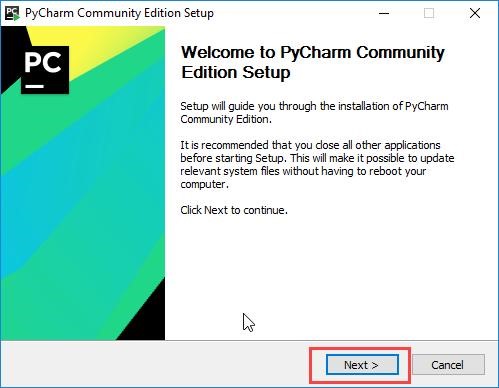
Krok 1. Pobranie instalatora pycharm ze strony

https://www.jetbrains.com/pycharm/download/#section=windows oraz wybranie darmowej

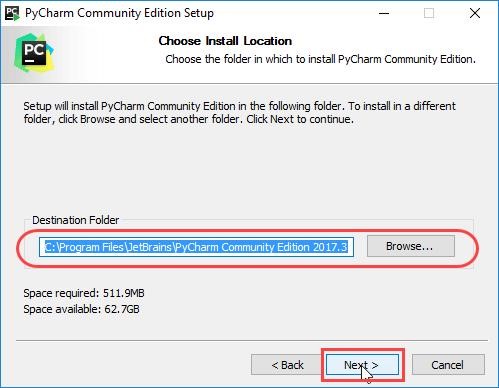
opcji



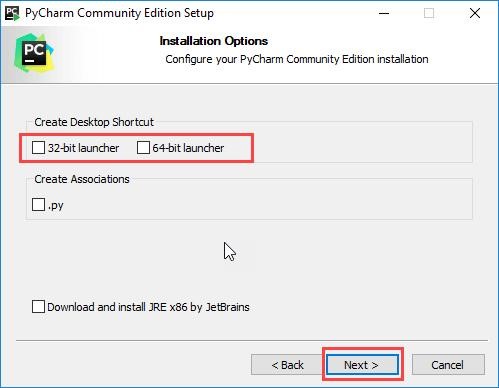
krok 2. Uruchomienie instalatora oraz postepowanie zgodnie ze zdjeciami

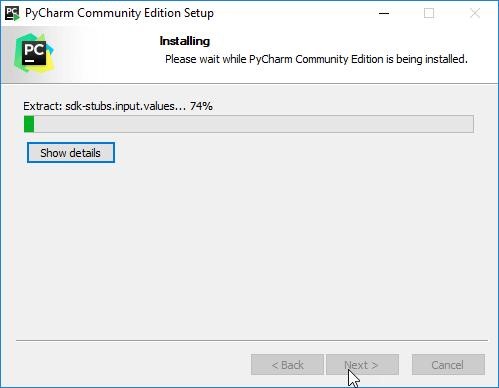
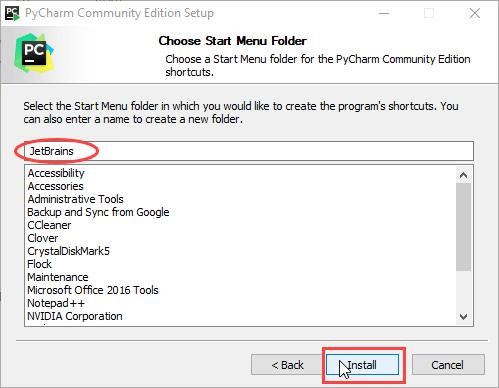


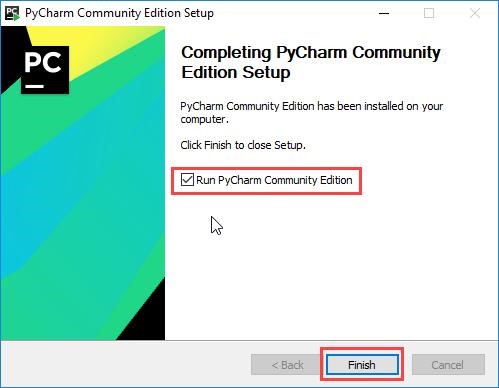
Folder instalacji pycharm moze zostac w domyslej lokacji



W tym okienku nalezy wybrac wersje 64bit lub 32bit zaleznie od systemu oraz mozna zaznaczyc "create associations .py" dzieki czemu kazdy program z koncowka .py zostanie uruchomiony w pycharmie

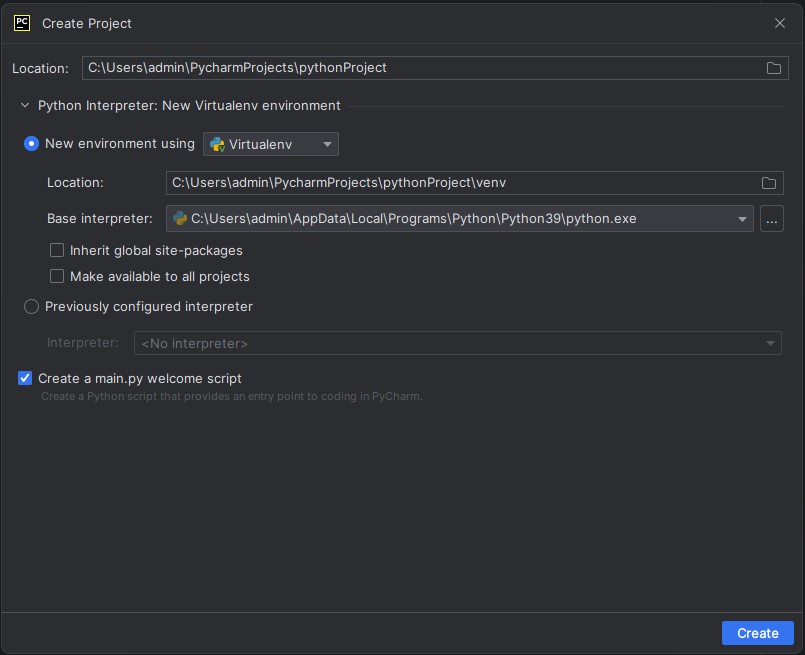




w tym momencie konczymy instalacje pycharma klikajac Finish

W tym miejscu mozemy wybrac folder z naszym projektem i naciskamy Create

polece? mog? by? wystarczaj?ce, natomiast dla du?ych projektów, bardziej zaawansowane



Zad 1

Narz?dzia programistyczne to oprogramowanie, które u?atwia tworzenie, testowanie i debugowanie kodu. Przyk?ady narz?dzi to edytory tekstu, narz?dzia do kontroli wersji, narz?dzia do tworzenia dokumentacji, narz?dzia do testowania kodu itp.

?rodowisko programistyczne (ang. Integrated Development Environment, IDE) to program, który zawiera zestaw narz?dzi programistycznych, takich jak edytor kodu, narz?dzie do debugowania, narz?dzie do tworzenia dokumentacji, narz?dzie do kontroli wersji itp. IDE pozwala na stworzenie kompleksowego i zintegrowanego ?rodowiska do pracy nad projektem.

W zale?no?ci od wielko?ci projektu, ró?ne narz?dzia i ?rodowiska programistyczne mog? by? bardziej lub mniej odpowiednie. Przyk?adowo, dla ma?ych projektów, edytor tekstu i linia programisty. Popularne narz?dzia i ?rodowiska programistyczne to Visual Studio, Eclipse, PyCharm, IntelliJ IDEA, Visual Studio Code, Sublime Text. Nie ma jednoznacznej odpowiedzi, które narz?dzie/?rodowisko programistyczne jest najciekawsze/najbardziej warto?ciowe dla rozwoju projektów programistycznych, bo to zale?y od indywidualnych potrzeb i preferencji.

# DZIAL 2

ZAD 1

Typ danych to okre?lenie rodzaju danych, które mog? by? przechowywane i przetwarzane przez komputer. ?rodowisko wywo?awcze Python przypisuje i przechowuje typy danych automatycznie, na podstawie warto?ci, które s? przypisane do zmiennej.

Istnieje mo?liwo?? zamiany jednego typu danych na inny, za pomoc? odpowiednich funkcji lub operatorów.

Proste typy danych zaimplementowane w j?zyku Python to:

liczby ca?kowite (int) - warto?ci ca?kowite, np. -5, 0, 100, itd. Ich rozmiar zajmuje 4 bajty w pami?ci operacyjnej

liczby zmiennoprzecinkowe (float) - warto?ci zmiennoprzecinkowe, np. 1.23, -3.14, itd. Ich rozmiar zajmuje 8 bajtów w pami?ci operacyjnej

?a?cuchy znaków (str) - ci?gi znaków, np. "hello", "abc123", itd. Ich rozmiar zale?y od d?ugo?ci ?a?cucha znaków

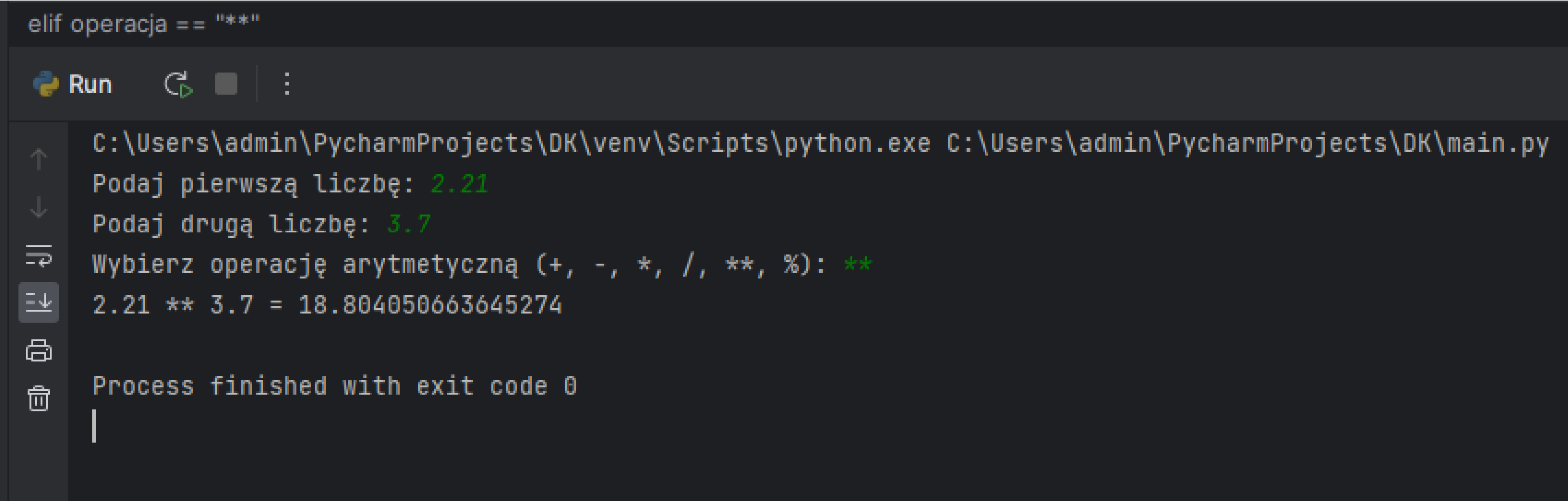
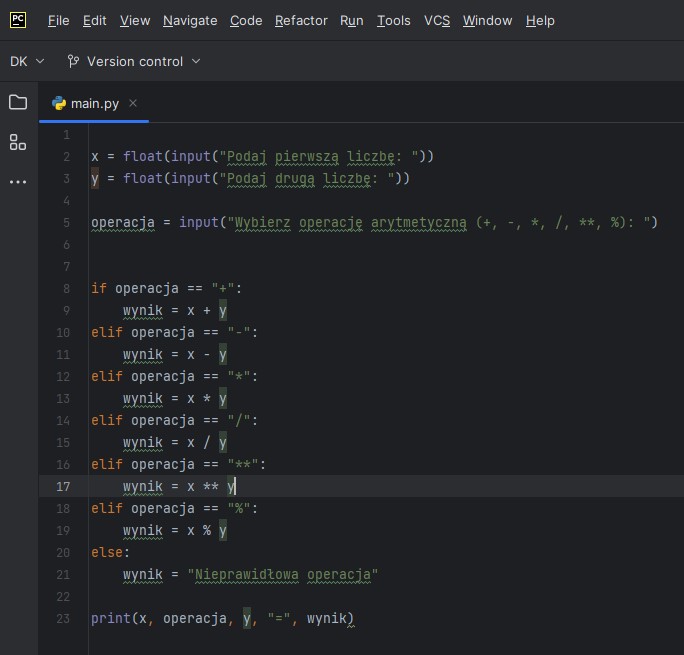
ZAD 2warto?ci logiczne (bool) - warto?ci true lub false. Ich rozmiar zajmuje 1 bajt w pami?ci operacyjnej

listy (list) - posortowana sekwencja obiektów, np. [1, 2, 3, 4] tuple (tuple) - niezmienne sekwencja obiektów, np. (1, 2, 3, 4) s?owniki (dict) - kolekcja par klucz-warto?? zbiory (set) - kolekcja unikalnych elementów

Rozmiar zajmowany przez te typy danych jest zmienny i zale?y od ilo?ci i warto?ci danych.Link do kodu z zadania

# https://github.com/frozuu/TEB\_PD/blob/main/main.py

DZIAL III



ZAD 1

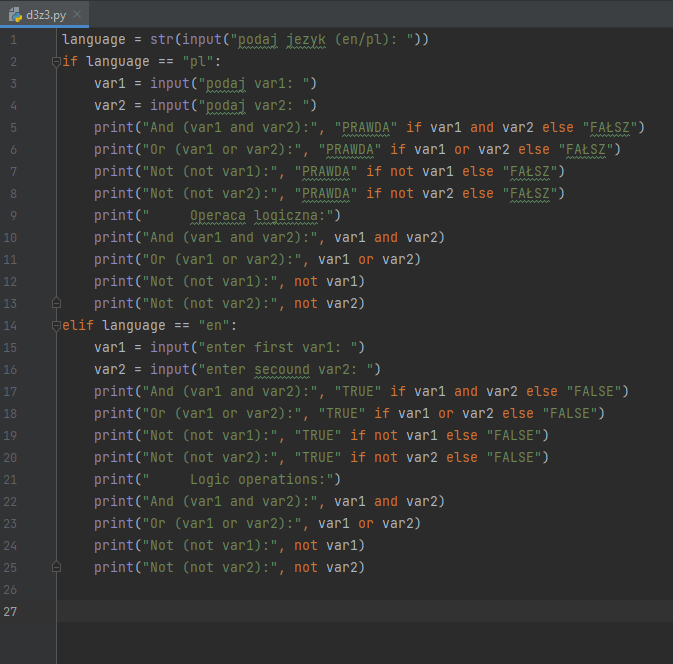
W jezyku Python istniej? rózne rodzaje rozwiaza? decyzyjnych, w tym:

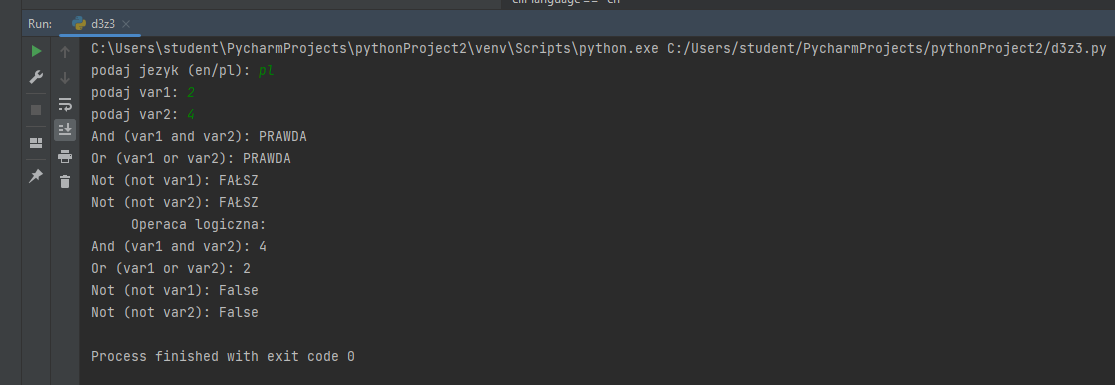
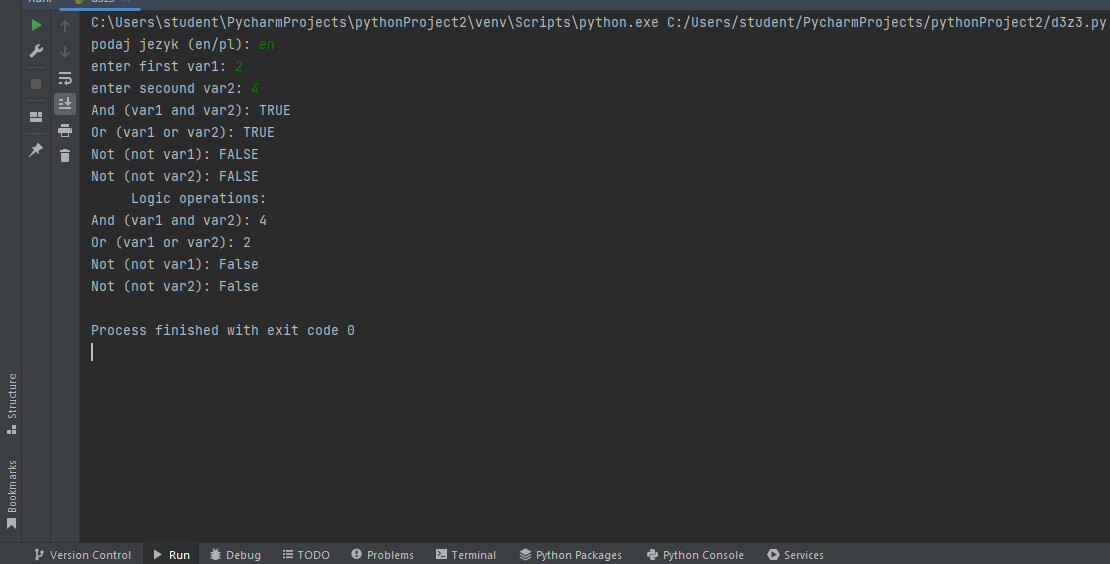
Instrukcje warunkowe: if, elif, else

P?tle: for i while

Instrukcje skoku: break, continue, return

Te rozwiazania pozwalaja na wykonywanie róznych operacji w zaleznosci od okreslonych warunków lub na powtarzanie operacji w petli.

ZAD 2 W jezyku Python nie ma natomiast natywnej implementacji rozwiazan takich jak case/switch lub goto. Mimo to, rozwiazania te moga byc zaimplementowane za pomoca innych instrukcji decyzyjnych, takich jak instrukcje warunkowe i petle.  
  
ZAD 2

[Powered by TCPDF (www.tcpdf.org)](http://www.tcpdf.org/)

[https:/github.com/frozuu/TEB\_PD/blob/main/d3z3.py](https://github.com/frozuu/TEB_PD/blob/main/d3z3.py)

Dzial IV

ZAD 1

Rola tworzenia pętli w oprogramowaniu polega na automatyzacji powtarzających się czynności. Dzięki pętom, programy mogą powtarzać sekwencje kodu wiele razy, bez konieczności ręcznego powtarzania tych samych instrukcji. Bez możliwości tworzenia pętli, programy byłyby znacznie bardziej skomplikowane i trudniejsze do utrzymania.

W języku Python istnieją dwa główne rodzaje pętli: for oraz while. Pętle for służą do iterowania po elementach sekwencji (np. listy, tuple, zakresu liczb), while służą do powtarzania kodu do momentu spełnienia określonego warunku.

Inne pętle, które można spotkać w innych językach programowania, takie jak pętle for-each, nie są dostępne w języku Python, jednak można je zastąpić za pomocą pętli for. Na przykład, pętla for-each w języku C#, która pozwala na iterowanie po elementach kolekcji, może być zastąpiona przez pętlę for w Pythonie, używając metody enumerate() lub iter().

ZAD 2