

## Podstawowe wytyczne do Programowania w Python

**Czysty kod w Pythonie** to taki, który jest łatwy do zrozumienia i utrzymania, a także jest prosty do rozwoju i modyfikacji. Oto kilka wskazówek, jak tworzyć czysty kod w Pythonie:

1. Nazwy zmiennych i funkcji powinny być opisowe i zgodne z konwencją nazewnictwa PEP 8. Na przykład, nazwy funkcji powinny być pisane małymi literami, z podkreśleniem między słowami (np. "moja\_funkcja"), a zmienne powinny mieć nazwy, które odzwierciedlają ich zawartość.
2. Unikaj długich funkcji i zagnieżdżonych bloków kodu. Powinny być one zawsze zwarte i rozbite na mniejsze części. Długość funkcji nie powinna przekraczać 20-30 linii kodu.
3. Unikaj powtarzania kodu i twórz funkcje, które można wykorzystać w innych częściach programu.
4. Staraj się unikać globalnych zmiennych, ponieważ mogą one powodować problemy związane z błędami i trudnościami w debugowaniu kodu.
5. Staraj się wykorzystywać komentarze do opisywania kodu, w szczególności, gdy wykorzystujesz bardziej złożone algorytmy. Komentarze powinny być jasne, zwarte i zawierać tylko niezbędne informacje.
6. Używaj bibliotek standardowych Pythona i unikaj pisanie własnych narzędzi, jeśli są one już dostępne.
7. Staraj się pisać kod z myślą o czytelności i łatwości zrozumienia, a nie optymalizacji i szybkości działania. Pamiętaj, że kod, który jest łatwy do zrozumienia, jest również łatwiejszy do modyfikowania i utrzymywania.
8. Pamiętaj o zasadzie DRY (Don't Repeat Yourself), czyli unikaj powtarzania kodu. Jeśli stwierdzisz, że kod jest powtarzalny, zrób z niego funkcję i wykorzystaj ją w innych częściach programu.
9. Staraj się stosować testy jednostkowe do weryfikacji poprawności działania kodu.
10. Przyjmuj spójne style formatowania kodu i dbaj o czytelne wcięcia, odpowiednią liczbę linii pomiędzy blokami kodu, stosuj spacje zamiast tabulatorów.

**PEP 8** to dokument definiujący standardy stylu kodowania dla języka Python. Oto niektóre z najważniejszych wytycznych zawartych w PEP 8:

1. Nazwy zmiennych, funkcji i klas powinny być pisane w stylu `snake_case` (np. `my_variable_name`).
2. Nazwy klas powinny być pisane z wielkiej litery (np. `MyClass`).
3. Wcięcia powinny być zawsze 4-spacjowe, a nie tabulacji.
4. Linie nie powinny przekraczać 79 znaków.
5. Importy powinny być umieszczane na osobnych liniach.
6. Puste linie powinny być stosowane w celu oddzielenia bloków kodu.
7. Stosuj jednoznakowe cytaty tylko do literałów znakowych, a nie do ciągów tekstowych.

```
my_variable = 'a'  
my_string = "To jest ciąg tekstowy."
```

8. Unikaj nadużywania operatorów skróconego zapisu (np. `+=` i `*=`) w przypadku złożonych wyrażeń.
9. Nie używaj pojedynczej litery `"l"` jako nazwy zmiennej, ponieważ może ona być łatwo pomyłona z cyfrą `"1"`.
10. Te wytyczne dotyczą jedynie kilku kwestii związanych z formatowaniem kodu w języku Python, ale są one kluczowe dla poprawnej konwencji i estetyki kodu.
11. Funkcje powinny mieć opisy `docstring`.

`Docstringi` są zapisywane w postaci ciągu znaków (najczęściej umieszczonych w trójkątach, np. `"""This is a docstring"""`) i są. Dzięki zastosowaniu `docstringów`, użytkownicy mogą szybko zrozumieć, jak korzystać z danego elementu i jakie parametry przyjmują.

Dobrze napisany `docstring` powinien zawierać opis funkcji, opis parametrów wejściowych, opis wartości zwracanych oraz przykłady użycia. Dzięki temu użytkownicy końcowi i inni programiści będą w stanie łatwo zrozumieć, jak korzystać z danego elementu i jakie parametry przyjmują.