

Ćwiczenie 1

Zaprojektuj hierarchię klas, która pozwala na wczytywanie, zapisywanie i przetwarzanie informacji tekstowych. Powinna spełniać następujące założenia:

1. `Text` jest kontenerem przechowującym dowolny rodzaj tekstu. Może zostać utworzony przez podanie mu tekstu do przechowania, lub przez wczytanie pliku tekstowego. Tekst może zostać zapisany do pliku. Umożliwia również poznanie całkowitej liczby liter w tekście oraz liczby wystąpień danego znaku w tekście.
2. Dla każdego z rozszerzeń `.txt`, `.md` istnieje osobna klasa, która umożliwia zapisanie tekstu do pliku z danym rozszerzeniem. Zwraca błąd przy próbie wczytania pliku z innym rozszerzeniem.

Zaimplementuj zaprojektowaną hierarchię klas w postaci Pythonowego modułu. Programując:

1. Trzymaj się PEP-8 oraz maksymalizuj czytelność kodu.
2. Stosuj typowanie wszędzie tam, gdzie to możliwe.
3. Moduł, klasy i funkcje powinny być opatrzone docstringami.
4. Moduł powinien służyć również jako skrypt. Powinien wówczas udostępniać następujące opcje:
 - (a) `-f`, `--file`: plik, który ma zostać wczytany (argument wymagany)
 - (b) `-o`, `--output`: plik, do którego ma zostać zapisany wczytany tekst (argument niewymagany)
 - (c) `-c`, `--count`: znak, którego liczbę wystąpień kod ma zwrócić i wypisać w terminalu jako efekt wywołania skryptu (argument niewymagany)

Ćwiczenie 2

Zaprojektuj hierarchię klas, która pozwala na reprezentowanie i przetwarzanie figur geometrycznych:

1. Każda figura ma obwód oraz pole.
2. Niektóre figury mają określoną liczbę boków oraz ich długość – to wielokąty.
3. Niektóre wielokąty są foremne.
4. Koło nie jest wielokątem.
5. Pole i obwód figury wynikają z jej kształtu (czyli liczby boków i ich długości). Wyjątkiem jest koło, które zamiast boków charakteryzuje się promieniem.

Zaimplementuj zaprojektowaną hierarchię klas w postaci Pythonowego modułu. Programując trzymaj się dobrych praktyk (jak w ćwiczeniu 1), oraz:

1. Uwzględnij, że informacje na temat wielokąta foremnego, koła oraz wielokąta nieforemnego mogą być przechowywane zupełnie inaczej.
2. Nie przejmuj się sprawdzaniem, czy z danego zbioru odcinków (boków) da się stworzyć wielokąt.
3. Nie implementuj obliczania pola i obwodu dla wielokątów nieforemnych.
4. Moduł powinien służyć również jako skrypt. Powinien wówczas udostępniać następujące opcje:
 - (a) `-s`, `--shape`: udostępnia opcje `circle` oraz `regular`, przy wpisaniu innej zwraca błąd (argument wymagany).
 - (b) `-n`, `--n-sides`: liczba boków (argument wymagany gdy `-s==regular`, można użyć `parse_known_args()`).
 - (c) `-l`, `--length`: długość boku bądź promienia, zależnie od wartości `-s` (argument niewymagany, przyjmujący domyślną wartość 1).

Skrypt na wyjściu wypisuje pole oraz obwód figury, jakiej parametry dostał na wejściu.