

# Lista 6

## Klasy i obiekty - ciąg dalszy.

Wstęp do programowania - laboratorium

Inżynieria systemów WIT PWr

1. (5 pkt) Napisz klasę `Restauracja`. Metoda `__init__()` w tej klasie powinna zawierać trzy atrybuty: nazwę restauracji oraz typ serwowanej kuchni a także liczbę obsłużonych klientów (ustaw na początek wartość domyślną 0). Utwórz metodę `opis_restauracji()`, która wyświetla powyższe informacje z konstruktora.  
Następnie utwórz metodę `otwarta_restauracja()`, która wyświetla informację czy restauracja jest otwarta. Kwestię warunku otwarcia można dowolnie zaimplementować wg własnej inwencji.  
Dodaj metodę `ustaw_liczbe_obsłużonych_klientow()`, która pozwoli ustalić dowolną liczbę obsłużonych klientów z poziomu interfejsu.  
Dalej, utwórz metodę `dodaj_liczbe_obsłużonych_klientow()`, która pozwoli o daną wartość zwiększyć liczbę obsłużonych klientów.  
W interfejsie stwórz przynajmniej trzy różne obiekty - instancje do klasy `Restauracja`, następnie wywołaj dwa atrybuty indywidualnie po czym wywołaj wszystkie metody dla każdego obiektu. Poeksperymentuj z ustawianiem, dodawaniem i modyfikowaniem liczby obsłużonych klientów z poziomu interfejsu.  
Utwórz klasę `Lodziarnia` dziedziczącą po klasie `Restauracja`. Dodaj nowy atrybut `smaki`, który będzie zawierał listę smaków lodów. Napisz metodę, która wyświetli te smaki. Utwórz instancję do nowej klasy w interfejsie i wywołaj tę metodę.
2. (5 pkt) Napisz klasę `Uzytkownik`. Utwórz w niej atrybuty: imię, nazwisko, próby logowania a także kilka innych typowych dla tworzenia konta użytkownika. Utwórz metodę `opisz_uzytkownika()`, która wyświetli powyższe zebrane informacje.  
Utwórz kolejną metodę `pozdraw_uzytkownika()`, która wyświetli spersonalizowane pozdrowienie użytkownika.

Napisz metodę `dodaj_probe_logowania()`, która zwiększa konstruktor próby logowania o 1.

Napisz następną metodę `resetuj_proby_logowania()`, która zmienia wartość prób logowania na 0. Utwórz kilka instancji reprezentujących różnych użytkowników i wywołaj wszystkie metody dla każdego z nich. Poeksperymentuj z dodawaniem liczby prób logowania a następnie zresetuj je. Admin jest specjalnym typem użytkownika. Napisz nową klasę `Admin`, która dziedziczy po klasie `Uzytkownik`. Dodaj atrybut `przywileje`, który zawiera listę zbudowaną ze stringów takich jak: może dodać post, może usunąć post, może zablokować użytkownika, etc. Utwórz instancję do nowej klasy `Przywileje`. Napisz oddzielną klasę `Przywileje`. Klasa powinna mieć jeden atrybut: `przywileje` będący listą przywilejów z klasy `Admin`. Napisz metodę `pokaz_przywileje()`, która wyświetla listę przywilejów administratora. Stwórz nową instancję w klasie `Admin` i wyświetl listę przywilejów z klasy `Przywileje`.