

Zadanie powtórkowe 6

A. Stwórz klasę `BankAccount` z następującymi elementami

- `__init__(self, number, balance=0)` - tworzy nowe konto o podanym numerze i początkowym stanie konta (`balance`).
- `deposit(self, amount)` - wpłaca zadaną ilość pieniędzy na konto.
- `withdraw(self, amount)` - wypłaca zadaną ilość pieniędzy z konta.
- `merge_to(self, other_account)` - przelewa wszystkie pieniądze z konta `self` na `other_account`.
- `__str__(self)` - zwraca napis z informacjami o koncie.

B. Stwórz klasę `SavingsAccount` dziedziczącą po `BankAccount` ze składowymi:

- `__init__(self, number, balance=0, interest_rate=0.05)` - nowy konstruktor, który oprócz numeru i początkowego stanu konta przyjmuje też stopę odsetek na jeden okres kapitalizacji. Wewnętrznie, konstruktor wywołuje konstruktor `BankAccount`, przekazując mu `number` i `balance`.
- `get_interests(self)` - wylicza odsetki, które powinny być naliczone podczas kapitalizacji. Odsetki to iloczyn stopy oprocentowania (podanej przy konstrukcji obiektu) oraz aktualnego stanu konta.
- `capitalize(self)` - dolicza do stanu konta odsetki, obliczone przy pomocy `get_interests`.

C. Stwórz klasę `CreditAccount` dziedziczącą po `SavingsAccount` ze składowymi:

- `withdraw(self, amount)` - robi to samo, co nadpisana metoda zdefiniowana w `BankAccount`, ale nie sprawdza, czy `amount` przekracza stan konta.
- `merge_to(self, other_account)` - to samo, co `merge_to` zdefiniowana w `BankAccount`, ale tylko pod warunkiem, że stan konta `self` nie jest ujemny (w implementacji używa `super()`, aby wywołać `merge_to` z klasy wyżej w hierarchii).
- `get_interests(self)` - oblicza wartość odsetek. Wartość wynosi 0, gdy stan konta jest ujemny, w przeciwnym wypadku jest naliczany identycznie, jak w `SavingsAccount`. Nie odnosi się jednak bezpośrednio do metod `SavingsAccount`.

D. Stwórz obiekty w typach określonych wyżej i przetestuj działanie metod.

E. Co się zmieni jak do klas z punktów 1-3 zmienne oznaczone będą jako prywatne? jakie metody trzeba dodać by uzyskać do nich dostęp? (wystarczy zastanowić się, bez wrzucania kodu)