

I. Utwórz klasę `Person` zawierającą publiczne pola:

- `String name`
- `int birthYear`

Utwórz obiekt klasy `Person` i przechowaj go w zmiennej `person`. Wykorzystaj możliwość dostępu do publicznych pól tej klasy i przypisz im literały lub wartości opisujące tworzoną osobę.

II. Przygotuj klasę `Person` definiującą pola jak w poprzednim zadaniu. Ponadto klasa będzie również definiować:

- dwuargumentowy konstruktor, inicjujący pola klasy;
- jednoargumentowy konstruktor, przyjmujący jako parametr `String name`, natomiast jako pole `birthYear` przypisujący wartość 1990;
- metodę `getName()` zwracającą wartość pola `name`;
- metodę `getAge()` zwracającą wiek osoby;
- statyczną metodę `getOlderPerson` przyjmującą w liście argumentów dwa obiekty klasy `Person` i zwracającą starszą osobę;
- statyczną metodę `getOldestPerson` przyjmującą jako argument tablicę obiektów klasy `Person` i zwracającą najstarszą osobę.

III. Utwórz konstruktor dla klasy `Person` który zainicjuje wszystkie jej pola. Ponadto utwórz metodę `show` wypisującą na ekran stan wszystkich pól. Działanie obu składowych klasy przedstaw w programie.

IV. Liczby zespolone składają się z części *rzeczywistej* i *urojonej* opisanych przez zmienne typu `double`. Napisz klasę `Cplx` implementującą:

- prywatne pola obu liczb
- dwuparametrowy konstruktor
- funkcję dodawania (`add(Cplx)`)
- funkcję odejmowania (`sub(Cplx)`)
- funkcję mnożenia (`mul(Cplx)`)
- funkcję zwiększającą część rzeczywistą o jeden (`inc()`)
- funkcję `show` wyświetlającą daną liczbę zespoloną w formacie $4 + 5i$

Działanie zdefiniowanej klasy przedstaw na przykładach przynajmniej trzech różnych liczb.