1. Stwórz klasę Osoba z polami imię i nazwisko. Następnie stwórz klasę potomną Student z polami rokStududiów, numerGrupy, numerAlbumu. Dodaj w obu klasach konstruktory domyślne i parametryczne.
2. W obu klasach stwórz metody WypiszInfo() wyświetlające wszystkie pola z klasy na konsoli (mamą to być tylko instrukcje Console.WriteLine() z parametrami).
3. Dodaj new aby pozbyć się ostrzeżenia (warning) w Visual Studio. Czemu takie rozwiązanie nie jest najlepsze?
4. Wykonaj rzutowanie w górę (Osoba student1 = new Student()) i wywołaj metodę WypiszInfo().
5. Usuń new (dodane w punkcie 3). Następnie do metody WypiszInfo() w klasie Osoba dodaj virtual, a w klasie Student override. Jaka jest różnica?
6. Zmodyfikuj metodę WypiszInfo() w klasie Student używając base.WypiszInfo().
7. Przerób klasę Osoba na abstrakcyjną.

---

1. Stwórz abstrakcyjną klasę Pozycja. Dodaj w niej pola tytuł (string), id (int), wydawnictwo (string), rokWydania (int) - wszystkie pola z mod. protected.
2. W klasie Pozycja dodaj konstruktor domyślny i parametryczny.
3. W klasie Pozycja dodaj deklarację abstrakcyjnej metody WypiszInfo().
4. Stwórz klasę Książka dziedziczącą z klasy Pozycja. Dodaj w niej prywatne pole liczbaStron (int). Następnie dodaj pole autor typu string (mod. private). Dodaj konstruktor domyślny i parametryczny. Następnie zaimplementuj metodę WypiszInfo().
5. Stwórz klasę Czasopismo dziedziczącą z klasy Pozycja. Dodaj w niej prywatne pole numer (int). Stwórz konstruktor domyślny i parametryczny oraz zaimplementuj metodę WypiszInfo().
6. Stwórz klasę Katalog. Dodaj w niej pole dzialTematyczny (string, private). Następnie stwórz pole referencyjne listaPozycji typu List<Pozycja> przechowujący pozycje z katalogu.
7. Stwórz konstruktory w klasie Katalog.
8. Następnie w klasie Katalog dodaj metodę DodajPozycje(Pozycja), która będzie dodawać na listę książkę lub czasopismo. Może warto ją przeciążyć?
9. Stwórz Interfejs IZarzadzanieKatalogiem. Dodaj w niej deklarację metod: WyszukajPoTytule(string), WyszukajPoId(int), WypiszWszystko().
10. Podepnij interfejs IZarzadzanieKatalogiem do klasy Katalog. Dodaj implementację metod.
11. Stwórz listę na obiekty z klas potomnych od klasy Pozycja i potestuj działanie metod.

---

1. Odszukaj kod z ćwiczeń nr 4, gdzie należało stworzyć grę.
    a. stwórz klasę na przechowywanie drużyny bohaterów (domyślnie 10 osób), drużyny mogą przechowywać bohaterów różnych typów (pamiętaj o właściwym rzutowaniu).
    b. dodaj metodę kopiująca bohaterów korzystającą z interfejsu IClonable.
    c. dodaj metodę na sortowanie bohaterów o żywotności lub liczbie punktów taktyki

d. klasę rodzica np. Bohater przerób na klasę abstrakcyjną, ogranicz niepotrzebne dublowanie kodu,
e. przetestuj działanie na kilku drużynach.