

C# Komplex Feladatleírás: Adatkezelő rendszer a hozzáférési módosítók használatával és külső osztályokkal

Feladat célja:

Készíts egy egyszerű adatkezelő rendszert, amely különböző hozzáférési módosítókat (public, private, protected) használ az adatok védelmére és a helyes rétegzett felépítés érdekében. A projekt tartalmazzon külső osztályokat is, amelyekkel az adatok kezelését modularizáljuk.

Feladat specifikáció:

1. Projekt leírása:

Készíts egy diákok adatait kezelő rendszert, amely a következőket tartalmazza:

- A diák nevét, életkorát és a tanulmányi eredményeit tárolja.
 - Egy külön osztály felelős az adatok feldolgozásáért (például átlag számításáért).
 - Hozzáférési módosítókat használ a megfelelő adatelérés érdekében.
-

2. Osztályok:

a) Student osztály (Diák adatainak tárolása):

- **Tulajdonságok (Fields):**
 - private string Name - A diák neve.
 - private int Age - A diák életkora.
 - protected List<int> Grades - A diák tanulmányi eredményei.
 - **Konstruktor:**
 - Paraméterben kapja meg a nevet, életkort és egy listát az eredményekről.
 - **Metódusok:**
 - public string GetName() - Visszaadja a diák nevét.
 - protected int GetAge() - Visszaadja a diák életkorát (csak az öröklődő osztályok férnek hozzá).
 - public List<int> GetGrades() - Visszaadja a diák jegyeit.
-

b) StudentProcessor osztály (Diák adatainak feldolgozása):

- **Feladata:**
 - A diák eredményei alapján átlagot számolni.
 - Adatok listázása.
 - **Metódusok:**
 - `public double CalculateAverage(Student student)` - Számolja ki a megadott diák eredményeinek átlagát.
 - `public void PrintStudentDetails(Student student)` - Írja ki a diák nevét, életkorát (ha elérhető), és eredményeit.
-

c) Program osztály (Alkalmazás belépési pontja):

- Hozzon létre legalább 3 diákot a Student osztály példányaival.
 - Használja a StudentProcessor osztályt az adatok kiírására és feldolgozására.
 - Példa egy átlag kiszámítására és kiírására.
-