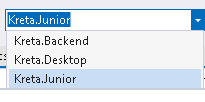
Egy szoftverfejlesztési csoportban dolgozik, amely egy iskolai adminisztrációs desktop alkalmazást készít. A fejlesztés során feladata a megfelelő adatstruktúra kifejlesztése (osztályok és repo), és a desktop alkalmazásban már megírt RestApi eléréseket és hozzájuk tartozó HttpService-ket használva adminisztrációs oldal elkészítése!

A fejlesztés során használni kell az OOP alapelveket, a tiszta kód alapelveket és a réteges szoftverfejlesztés alapelveit!

**Student feladat**

**Első munkamenet: I. Osztályok fejlesztése**

A feladatot a Kreta.Junior projektbe oldja meg. Más projektek kódját nem használhatja! Teszteléshez állítsa be, hogy a futtatott projekt a Kreta.Junior projekt legyen.

****

1. Talál a model rétegben egy osztályt AwardedStudent (tanulmányi eredményeik miatt alapítványi támogatásban, ösztöndíjban részesített diákok) néven azon diákok számára, akik ösztöndíjat kapnak!

2. Az ösztöndíjas diákoknak tudjuk a nevét, életkorátt és az ösztöndíj értékét (Ft-ban) és hogy az ösztöndíj egyszeri juttatás vagy havonta járó juttatás!

3. Készítse el a szükséges konstruktorokat!

4. Készítse el az osztályt szöveggé alakító metódusát, úgy, hogy példányosítás után az adatok megjelenítésekor a pontosan következő szöveg jelenjen meg:

"Ügyes Tünde 17 éves és 1525 Ft ösztöndíjat nyert! Az ösztöndíj egyszeri juttatás!"

Jelenítse meg a fő programmal a minta szerint a diák adatait! Ha az ösztöndíj havi juttatás, akkor a szöveg vége „Az ösztöndíj havi juttatás!” legyen.

5. Hozzon létre még egy díjazott diákot aki Dolgos Dénes, 18 éves és 5000 Ft egyszeri juttatásba (ösztöndíjban) részesül. Vezérlési szerkezettel jelenítse meg a fő programban, hogy melyik diák az idősebb! A kód kimenetei a következők közül lehetnek:

"Ügyes Tünde idősebb, mint Dolgos Dénes!"

"Dolgos Dénes idősebb, mint Ügyes Tünde!"

"A két diák ugyan annyi éves!"

A kiírásban a nevek változó adatok!

6. Készítsen az osztályban csak olvasható peropty-t, amely akkor ad igaz értéket, ha a támogatott diák már betöltötte a 18. életévét!

7. Készítsen az osztályban statikus metódust IsYounger néven, amely megadja, hogy a metódus paraméterében megadott első diák fiatalabb-e a paraméterben megadott második diáknál! A kódot TDD módszerben készítse el! A tesztesetek megadására készített táblázatba, amely tartalmaz legalább három tesztesetet az elvárt kimenetelekkel! Készítse el a teszteket! Írja meg az IsYounger metódus kódját úgy, hogy egyetlen teszteset se legyen hibás!

**Második munkamenet: II. Adattárak (repository) fejlesztése**

A feladatot a Kreta.Junior projektbe oldja meg. Más projektek kódját nem használhatja! Teszteléshez állítsa be, hogy a futtatott projekt a Kreta.Junior projekt legyen.

8. Készítsen el egy új osztály AwardedStudent Repo néven, amelyben több ösztöndíjjal rendelkező diák tárolására alkalmas adatstruktúrát használjon!

Bővítse úgy a megfelelő osztályt, hogy az ösztöndíjasok nemét is ismerjük!

Egészítse ki az ösztöndíjjal rendelkező diákok osztályát egy olyan property-vel amely tartalmazza a diák osztályát is! Pl. „9.a”

9. Adja hozza Üveges Tünde és Dolgos Dénes-t a repóhoz!

Üveges Tünde a 9.a osztályba jár és nő.

Dolgos Dánes a 9.a osztályba jár és férfi.

Adja hozzá a következő adatokat a repóhoz:

Szorgalmas Szonja 16 éves, 6527 Ft ösztöndíjat kap havonta, "9.b" osztályba jár és nő!

Mindenttudó Misi 16 éves, 6514 Ft ösztöndíjat kap havonta, "9.b" osztályba jár és férfi!

A repository kódját úgy készítse el, hogy alkalmazza a réteges programozás elvét, vagyis annak adataihoz, csak a publikus metódusin keresztül lehessen hozzáférni. Azon feladatok megoldásai, amelyek sértik ezt az elvet, nem kerülnek pontozásra!

Fontos tudni, hogy a havi ösztöndíjat a diákok 10 hónapig kapják egy iskolai évben!

9. [Havi ösztöndíjak átlaga]

Tekintse az egy hónapban kifizetett havi ösztöndíjakat (nem egyszeri juttatásokat (ösztöndjakat)! Határozza meg ezen kifizetések havi átlagát két tizedes jegy pontossággal. Az eredményt „A havi ösztöndíjak átlaga xx.xx Ft” formában írja ki a képernyőre!

10. [Nemenként az ösztöndíjasok száma]

Írjon megoldást a kódban, amely meghatározza nemenként az ösztöndíjasok számát. Ha lehetséges a feladatot egy metódus segítségével oldja meg! Az eredményt a következő formátumba jelenítse meg:

Ösztöndíjasok megoszlása nemenként:

A fiú diákok száma x fő.

A lány diákok száma x fő.

11. [Egyszeri juttatások egy évben]

Jelenítse meg a fő programban felhasználóbarát módon, hogy *egyszeri* jutatásokból (ösztöndíjból) mennyi kifizetés történt egy iskolaévben

12. [Iskolaévben kifizetett ösztöndíjak összege]

Jelenítse meg felhasználóbarát módon a fő programban, hogy egy iskolaévben összesen mennyit kell fizetni az ösztöndíjasoknak (egyszeri juttatás (ösztöndíj) és havi ösztöndíj együtt)!

13. [Adóigazolás]

Azoknak, akik már betöltötték a 18. életévüket adóigazolást kell kiadni. Írja ki a képernyőre, a diák nevét és mennyi az ösztöndíjük! Az adatokat a képernyőn a fő program segítségével írja ki! A metódust úgy írja meg, hogy az ne string visszatérési értékkel rendelkezzen!

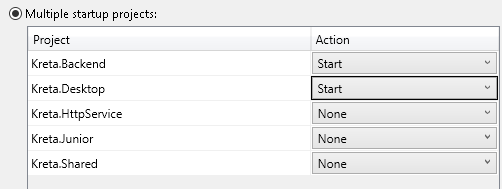
14. [Legfiatalabb diák]

Fejlesszen metódust GetYoungestAwardedStudentName() néven amelyet TDD módszerrel fejlesszen ki. A metódus a legfiatalabb ösztöndíjas nevét adja vissza! Legalább három tesztesetet esetén készítse el a tesztesetek táblázatát! Írja meg a metódus kódját úgy, hogy a tesztesetek között ne legyen olyan, amelyik elbukik!

**Harmadik munkamenet: III. Adatok kezelésére alkalmas desktop adminisztrációs oldal fejlesztése**

**A feladat tanárok és szülők esetén is értelemszerűen megoldhatók!**

A feladat megoldását a Kreta.Junior alkalmazás mellett lévő többi projektet használva tudja megoldani. Kódot írni a Kreta.Desktop projektbe kell, de a többi kódot felhasználhatja vagy igény szerint fejlesztheti. Állítsa be teszteléshez a szükséges futtatandó projektet:



A megoldás során használja a Kreta.Shared osztályban kifejlesztett Student osztályt!

A backenden ellenőrizze a meglévő RestApi hívásokat!

***User story***

*Én a Kréta alkalmazás adminisztrátora  
szeretném látni a rendszerben megtalálható diákokat,  
azért hogy pontos képem legyen a diákok adatairól!*

*Ha az adminisztrátor rákattint az Iskolapolgárok/Diákok menüpontra  
jelenjen meg az összes diák adata!*

**Tehát a feladat:**

1. Nyissa meg a desktop alkalmazás view és view model rétegében a megfelelő állományokat! A service réteg megfelelő metódusával eléri az osztályfőnökök listáját. Írjon kódot, amely az „Iskolapolgárok/Diákok ” menüpontban megjeleníti az összes diák összes adatát egy arra alkalmas vezérlőben!

***User story***

*Én a Kréta alkalmazás adminisztrátora  
szeretném a kiválasztott diákok adatait módosítani  
azért, hogy az adataik pontosak és javíthatók legyenek.*

*Ha az adminisztrátor megjelenítette a diák adatait és rákattint egy diákra,  
akkor annak adatai jelenjenek meg vezérlőkben úgy, hogy azokat módosítani lehessen.*

*Ha az adminisztrátor módosította a kiválasztott diák adatait,   
a Mentés gombra kattint a módosított adatok mentésre kerüljenek  
és az diákok listájában már a módosított adatok jelenjenek meg!*

**Tehát a feladat:**

1. Készítsen űrlapot az adatok módosítására!

2. Ha felhasználó rákattint egy diákra a már elkészített listában, az űrlapon jelenjenek meg annak adatai!

3. A Student RestApi címen PUT kéréssel lehetséges diákok adatának módosítása.

Használja a service réteg megfelelő metódusát a funkció eléréséhez! Ha a felhasználó a mentés gombra kattint, az adatokat mentse el és frissítse a mentett adatokkal a listát!

4. A felhasználók szerint nagyon zavaró, hogy módosítás után nem a módosított diák lesz kijelölve a listában! Hajtson végre fejlesztést a desktop alkalmazásban, hogy a diák módosítása után a módosított diák adata legyen a listában kiválasztott és így az űrlapon is az ő adatai legyenek!

A következő feladat az előző desktop feladatoktól függetlenül működik!

A feladat full stack feladat, a backend, service réteget is meg kell írni!

***User story***

*Én a Kréta alkalmazás adminisztrátora  
szeretném látni, hogy a diákok közül, hányan kinevezett és hány lány és fiú  
azért, hogy az iskoláról szóló jelentésben az adatot meg tudjam jeleníteni és, hogy vezetői döntéseket tudjak hozni.*

*Ha az adminisztrátor az Iskolapolgárok/Diákok menüpontra menüpontot kiválasztotta és a  
 „Statisztikai adatok” gombra kattint,  
akkor jelenjenek meg felirattal az lány és fiú diákok száma „fő” mértékegységben egymás alatt.*

**Tehát a feladat:**

6. Jelenítse meg az Iskolapolgárok/Diákok menüpontban arra alkalmas vezérlőben, hogy a diákok közül hányan lányok és hányan fiúk. A megjelenítéskor jelölje melyik adat melyik értékre vonatkozik és használjon mértékegységet a számok után!

**Full stack feladat**

1.

*Én a Kréta alkalmazás adminisztrátora  
szeretném szűrni a diákokat név, születési idő és nem szerint*

*hogy tisztába legyek az iskola diákjainak adataival!*

*Ha én a Kréta alkalmazás adminisztrátora az Iskolapolgárok/Diákok menüpontot választom*

*akkor a lenyitható űrlapon a „Szűrés” gombra kattintva szűrni tudok a diák nevére, születési idejére és nemére*

*valamint a szűrést visszavonni a „Szűrés visszavonása” gombra kattintva szeretnék!*

Minta a szűrés űrlaphoz:

**

*2.*

*Én a Kréta alkalmazás adminisztrátora  
szeretnék statisztikai adatokat lekérdezni a rendszerből,  
hogy tisztába legyek az iskola statisztikai adataival és vezetői döntéseket tudjak hozni.*

*Ezért ha a Vezérlőpult menüpontot választom ki  
akkor megjelenik a képernyőn, hogy hány szülő van a rendszerben, mennyi a férfi és női szülők száma  
valamint osztályönként hány diák van!*