

Junior Java Backend projektfeladat

I. feladat

Java programozási nyelv alapjai

A feladat01 package-ben termékek kategorizálását és árképzését végző alkalmazást kell befejeznie. A kategorizálást végző alprogram paraméterben kapja meg a kategóriakódot, majd a kódnak megfelelően szolgáltat egy szöveges eredményt. A lehetséges kódok és a hozzájuk tartozó szövegezés:

- 1 = "I. osztályú termék"
- 2 = "II. osztályú termék"
- 3 = "III. osztályú termék"

Feladatok:

- a) A teszt01 package-ben található egységteszt nem fut le sikeresen, javítsa/bővítse a megfelelő metódust!
- b) Írjon egy újabb metódust, ami két paramétert vesz át, a kategóriakódot és a termék alapárát. A kód a végső ár meghatározásához szükséges, első osztályú termék 100%, második osztályú 90%, harmadik osztályú 80%-a az alapárnak. Az alprogram az így számított értéket szolgáltatja a hívónak.
- c) A program kérje be a felhasználótól a termék kategóriakódját és az alapárát (formai ellenőrzés nem szükséges) és hívja meg a két metódust, majd írja is ki a képernyőre az eredményt, például, ha a kategóriakód = 2 és az alapár = 1500, akkor a kiírás:
II. osztályú termék, alapár 1500 Ft, aktuális ár: 1350 Ft
- d) Írjon egységtesztet az árképző metódus működésére vonatkozóan!

II. feladat

Java objektumorientált programozás

A feladat02 package-ben egy virtuális futballmérkőzést lebonyolító és közvetítő alkalmazást kell megvalósítania. A mérkőzés adatait Merkozes osztály tárolja, ami rendre a hazai csapat neve, vendég csapat neve, a mérkőzés időtartama (perc), hazai gólok száma és vendég gólok száma. Példányosításkor a konstruktor paraméterben kapja meg a csapatok neveit, míg a többi adattagot beállítja (időtartam = 90, hazai gólszám = 0, vendég gólszám = 0).

Feladatok:

- a) A teszt02 package-ben lévő egységteszt nem fut le sikeresen. Bővítse a konstruktort a feladatnak megfelelően!
- b) Merkozes osztály implementálja a void goool(String melyikCsapatSzerezte) metódust előíró interfészt. Az előírt alprogram növelje eggyel a paraméterben átadott csapat góljainak számát. A szükséges interfészt is hozza létre.
- c) Tetszőleges adatokkal hozzon létre egy mérkőzést. A mérkőzés időtartamával megegyező alkalommal futtasson ciklust, ahol a ciklusváltozó jelzi, hogy a mérkőzés hányadik percben jár. Iterációnként állítson elő véletlenszámot 1-25 között. Mérkőzés objektum goool metódusa kerüljön meghívásra a megfelelő paraméterrel, ha 10-es (hazai gól) vagy 20-as (vendég gól) érték generálódik.
- d) A képernyőn jelezze a gólokat, a gólszerző csapat és az aktuális perc kiírásával. Szolgáltasson végeredményt a Merkozes osztály toString() metódusának meghívásával! Egy lehetséges példakimenetel:

A mérkőzés Hazai SC és Vendég SE között megkezdődött:
Góóól! szerezte Vendég SE a 3. percben
Góóól! szerezte Hazai SC a 44. percben
Góóól! szerezte Vendég SE a 84. percben
A meccs végeredménye: Hazai SC - Vendég SE 1 – 2

- e) Írjon egy tesztet, amely azt ellenőrzi, hogy goool metódus jól állítja be Merkozes objektum megfelelő adattagját!

III. feladat

Java kollekciók

A feladat03 package-ben vizsgázók adatait tároló alkalmazást kell megvalósítania. Vizsgázókról tárolandó adatok: vizsgázó neve, email címe, képző intézmény neve. Az adatok bekérését a felhasználótól AdatKezeles osztály adatBekeres() metódusa végzi. Az osztály rendelkezik egy azonositoGeneralas() alprogrammal is, ami minden vizsgázóhoz rendel egy azonosítót, melynek felépítése: "Vizsgazo"+egy futó sorszám (ami AdatKezeles osztály statikus adattagja).

Feladatok:

- a) A teszt03 package-ben lévő egységteszt nem fut le sikeresen. Javítsa/bővítse a generáló metódust és az osztályát!
- b) Valósítsa meg a vizsgázók adatait tároló osztályt, a kiírásnak megfelelően!
- c) Hozzon létre 4 példányt az osztályból, az adatokat a felhasználótól kérje be, AdatKezeles objektumon keresztül. Minden vizsgázóhoz rendeljen azonosítót, ami egyben a tároló adatszerkezet kulcsa legyen. A hozzá tartozó érték pedig az éppen létrehozott Vizsgazo objektum. A kulcs-érték párt megvalósító adatszerkezetet eredményként szolgáltatja a bekérést végző alprogram.
- d) A főprogramban listázza ki az összes vizsgázó minden adatát, az azonosítójával együtt, majd állapítsa meg és írja ki a képernyőre, hogy érkezett-e vizsgázó a "ProfiSuli" intézményből! A logikai eredményt főprogram külön metódusa szolgáltassa, ami paraméterben kapja meg a keresett intézmény nevét és magát az adatszerkezetet.
- e) Ez utóbbi metódusra vonatkozóan írjon egy új tesztet!

IV. feladat

Fájlkezelés Javában

A feladat04 package-ben egy kábeltévé szolgáltató csatornahely kiosztását feldolgozó alkalmazást kell megvalósítania. Az adatok fájlból, normál vesszővel elválasztva érkeznek (megnevezés, kategória, nyelv, csatornahely, alapítás éve). Az UTF-8-as kódolású fájl (kiosztas.csv) a projekt mappájában található és fejléctet nem tartalmaz. A fájl beolvasását FajlKezeles osztály fajlBeolvas metódusa végzi, ami a fájl egyes sorainak adatai alapján létrehozott Csatorna osztály objektumait tároló listát szolgáltatja eredményül.

Feladatok:

- a) A teszt04 package-ben lévő egységteszt nem fut le sikeresen. Bővítse a megfelelő osztályokat/metódusokat!
- b) Listázza ki a képernyőre a 2010 után alapított csatornák adatait!
- c) Írja is ki ugyanezeket az adatokat a "csatorna2010utan.csv" fájlba! A fájlbaírást szintén FajlKezeles osztály végezze. Írja ki a képernyőre, hány csatorna felelt meg ezen követelménynek!
- d) Bővítse az alkalmazást egy tetszőleges tesztet létrehozásával!

V. feladat

Adatbáziskezelés Javában

A feladat05 package-ben vasútvonalak adatait kezelő ablakos (Swing) alkalmazást kell megvalósítani. Az alkalmazás adatbázissal (mysql motor) szinkronizál. A projekt könyvtárában található vasutvonalak_telepit.sql fájl futtatásával létrehozható az adatbázis és a hozzá tartozó tábla, adatok inzertálásával együtt.

A tábla oszlopai: vonalszám, indulási állomás, célállomás, távolság (km), menetjegy alapára (Ft). A program induláskor beolvassa az adatokat az adatbázisból, és egy táblázatban meg is jeleníti.

Feladatok:

- a. Futtassa az sql fájlt az adatbázis létrehozásához!
- b. Az adatbázis kapcsolódáshoz szükséges mysql driver szintén a projekt mappájában található, melyet állítson be a projekthez!
- c. A teszt05 package-ben lévő egységteszt nem fut le sikeresen. Bővítse a megfelelő metódust vagy metódusokat!
- d. Hozzon létre egy beviteli mezőt és egy gombot a táblázat alatt, "Árváltoztatás" felirattal! A gomb megnyomásakor a beviteli mezőben megadott %-os értékkel kell megváltoztatnia az egyes vasútvonalak alapárát. Szinkronizáljon adatbázisba, illetve a táblázatban is az aktualizált értékek jelenjenek meg!
- e. Bővítse az alkalmazást egy tetszőleges tesztet létrehozásával!