

# PNy Java ZH gyakorlati feladat 2022.06.02. 13:00

<b>Határidő</b> Nincs megadva határidő	<b>Pont</b> 20	<b>Kérdések</b> 1
<b>Elérhető</b> jún 2, 13:15 - jún 2, 15:00 körülbelül 2 óra		<b>Időkorlát</b> Nincs
<b>Engedélyezett próbálkozások</b> Korlátlan		

Ezt a kvízt ekkor zározták: jún 2, 15:00.

## Próbálkozások naplója

	Próbálkozás	Idő	Eredmény
LEGUTOLSÓ	<a href="#">1. próbálkozás</a>	79 perc	0 az összesen elérhető 20 pontból *

\* Néhány kérdés még nem lett értékelve

Ezen próbálkozás eredménye: **0** az összesen elérhető 20 pontból \*

Beadva ekkor: jún 2, 14:35

Ez a próbálkozás ennyi időt vett igénybe: 79 perc

### 1. kérdés

Még nincs értékelve / 20 pont

## Programozási Nyelvek Java ZH, 2022.06.02. 13:00 turnus

### Feltételek

A kódról.

- Feltételezhető, hogy meghíváskor érvényes adatokat kapnak a konstruktorok és metódusok: pl. a megkapott fájlnevek értelmesek, a hivatkozások nem `null` értékűek stb.
  - Kivétel: a szöveg explicit kérheti bizonyos adatok ellenőrzését.
- A kód legyen jó minőségű, a tanultaknak minél inkább megfelelő.
  - A mezők láthatósága mindig a lehető legszűkebb legyen.
  - A metódusok legyenek nyilvános láthatóságúak.
  - Legyen jól működő egységbe zárás: ne legyen adatszivárgás.
- A megoldás elkészítéséhez a [Java API dokumentáció](https://kitlei.web.elte.hu/exammaterials/java-api/) (<https://kitlei.web.elte.hu/exammaterials/java-api/>) használható, más segédeszköz nem.
  - Ha valami nem egyértelmű, érdemes az oktatóktól kérdezni.
  - A részfeladatokat a leírt sorrendben kell megoldani.

A beadásról.

- A félreértések elkerülése végett: a ZH napján tilos megosztani az elkészített megoldást bármilyen módon.
- Az elkészített megoldás forrásfájljait **zip** formátumba csomagolva kell feltölteni a Canvasbe.
  - A **zip** csak a **java** forrásfájlokat tartalmazza.
  - A ZH végén kb. **10 percet érdemes fenntartani** a kód tisztázására, fordíthatóvá tételére, tömörítésére, beküldésére.

## Feladat

### 1. Cégek -- **Company**

4 pont

Hozza létre a **commerce.company** csomagban a **Company** absztrakt osztályt.

1. A típusnak legyen egy **establishmentYear** ("alapítás éve") egész, és egy **name** szöveges mezője. A mezők láthatósága legyen ún. *package-private*.
  1. A mezőkhöz készítsen gettert.
  2. A mezők értéke az objektum létrejötte után többé nem változhat meg.
2. A konstruktor a fenti sorrendben várja a paramétereket, és beállítja a mezők értékét. Feltételezhető, hogy a paraméterek helyesek, ellenőrzésük nem szükséges.

### 2. Tömegközlekedési vállalat, alapok -

- **PublicTransportCompany**

6 pont

Készítse el a **commerce.company** csomag **PublicTransportCompany** típusát a **Company** típusból leszármaztatva.

1. A típusnak legyen egy **places** mezője, amely egy (potenciálisan nagy méretű) szöveges listában tárolja azon helyeket, ahol a tömegközlekedési vállalat működik (pl. **Budapest, Szeged, Gyor, Debrecen, Kaposvar**). A mező, miután értéket kapott, már nem módosulhat. Készítsen a mezőhöz gettert.
2. A típus egyetlen, publikus konstruktora három paramétert vár: előbb a szülő típus konstruktorának megfelelő paramétereket (az ottani sorrendben), majd utánuk a **places** mezőben tárolandó adatszerkezetet.
3. A típus szöveges reprezentációját -- a szokásos módon -- az alábbi mintájára készítse  
el: **"Volanbusz:1927:Budapest,Szeged,Gyor,Debrecen,Kaposvar"**
  1. A cég neve.
  2. Kettőspont.
  3. A cég alapítási éve.

4. Kettőspont.

5. Azon helyek listája, ahol a cég működik: a `places` elemei egymástól `,` karakterrel (vesszővel) elválasztva (az utolsó elem után nincs vessző).

### 3. Tömegközlekedési vállalat, factory -

- `PublicTransportCompany`

6 pont

1. Készítsen egy nyilvános `createFromString` osztályszintű metódust, amely egy, a típus szöveges reprezentációjának megfelelő szöveget kap, és visszatér az ebből megkonstruálható `PublicTransportCompany` objektummal. (Pl. `createFromString("Volanbusz:1927:Budapest,Szeged,Gyor,Debrecen,Kaposvar")`.)

Tipp: `String.split`, lásd a Java API-ban.

2. A szöveges paramétert ellenőrizni nem szükséges (feltételezhető, hogy a paraméter minden szempontból helyes is és "értelmes" is).

### 4. Főprogram -- `Register`

4 pont

Készítsen főprogramot az `app` csomagban `Register` néven, amely parancssori paraméterekben cégek szöveges reprezentációját (lásd fentebb) várja: egy paraméterben egy szöveges reprezentációt.

Példa:

```
$ java Register "Volanbusz:1927:Budapest,Szeged,Gyor,Debrecen,Kaposvar" "MAV-ST  
ART:1868:Budapest,Ukk,Apc"
```

1. A főprogram egy `companies` halmaz típusú mezőben `Company` típusú objektumokat fog tárolni, amelyeket a parancssori paraméterekből konstruál (feltételezhető, hogy vannak paraméterek, és hogy mindegyik paraméter egy `PublicTransportCompany` -- helyes -- szöveges reprezentációját tartalmazza):
2. A főprogram végigiterál a paramétereken, és mindegyik alapján létrehoz egy `PublicTransportCompany` típusú objektumot.
3. A főprogram ezt követően a sztenderd kimenetre egy új sorba kiírja a halmazban található elemek számát.
4. A főprogram ezt követően a sztenderd kimenetre soronként kiírja az eltárolt cégek szöveges reprezentációját (egy sorba egy objektum reprezentációját), majd visszatér.

↓ [zh-bb.zip \(https://canvas.elte.hu/files/1781814/download\)](https://canvas.elte.hu/files/1781814/download)

Kvízeredmény: **0** az összesen elérhető 20 pontból