

Néhány észrevétel



ÉSZREVÉTELEK ZH-JAVÍTÁS KÖZBEN

Névkonvenciók:

csomag: csupa kisbetű

osztály: nagy kezdőbetű

metódus, változó: kis kezdőbetű

Ezen belül: „camel case”.

Java-ban nem szokás az aláhúzás a nevekben.

Legközelebb szigorúan veszem.

Miért? Mert pl. a J2EE ki is használja ezt a konvenciót.

De ezért is: EloadasPanel.beolvasas();

ÉSZREVÉTELEK ZH-JAVÍTÁS KÖZBEN

Egy alaposztályban:

```
private final int MARKETING_SZORZO = 2;
```

illetve:

```
if(idoTartam > Global. IDO_LIMIT)...
```

Jó-e?

Alaposztályban NEM hivatkozhatunk a Global osztályra és NEM égethetjük be az adatokat.

Helyesen: statikus adattag + setter.

ÉSZREVÉTELEK ZH-JAVÍTÁS KÖZBEN

Beolvasás:

Ha nem interfész implementálásaként írjuk meg, akkor nem sok értelme van külön beolvasó osztályt írni, ekkor a panelen is megírhatjuk az olvasást.

NE tárolják duplán az adatokat! NEM kell lista is és modell is.

Végiggondolandó: Megjelennek-e az adatok egy lista-felületen. Ha igen, akkor modellbe kell rakni, ha nem, akkor listába.

A rendezéshez valóban kellett, de csak lokálisan.

ÉSZREVÉTELEK ZH-JAVÍTÁS KÖZBEN

Még mindig beolvasás:

Ha a beolvasás után azonnal rendezni akarunk, akkor nem modellbe, hanem listába kellett volna olvasni, ezt rendezni, majd átrakni modellbe. Teljesen fölösleges a modellből listába lépés.

Sokszor végiggondolás nélkül próbálják ugyanazt írni, mint órán - ez bizonyos szinten még érthető, de tovább kellene lépni.

ÉSZREVÉTELEK ZH-JAVÍTÁS KÖZBEN

Még mindig beolvasás – egy példa:

```
public List<Tanora> tanoraLista() throws Exception {  
    List<Tanora> tanorak=tanoralista;  
    ...  
    return tanorak;  
}
```

Pedig írt egy List<Eloadas> eloadasLista() metódust, amelyben feltöltötte a tanoralistát, csak épp meg lehet hívni a tanóralista metódust az előadáslista metódus hívása nélkül is. (Így is történt).

Oké, akkor hívjuk meg a tanoraLista() metódusban az eloadasLista() metódust - ekkor viszont kétszer is végigolvassa az adatfájlt, rossz hatékonyság. (Ilyen is volt.)

EZ MÁR NEM ZH-JAVÍTÁS, HANEM TOVÁBBLÉPÉS

De még mindig beolvasás ☺:

Az MVC szemlélet szerint egy külön beolvasó osztályban mindig listába olvassuk az adatokat, és a panelen rakjuk át valamilyen modellbe.

Korrekción:

Sajnos .jar-ból nem érhető el a ...toURI() módon megadott fájl, ahhoz, hogy .jar-ból is tudjuk futtatni, az inputstream-en keresztül kell elérnünk a fájlt.

Bocsánat ☹