

1. Absztrakt Személy Osztály

Az Személy osztály egy absztrakt szülőosztály, amely az egyetemi rendszerben szereplő összes személy közös tulajdonságait és viselkedését definiálja. Ez az osztály nem példányosítható közvetlenül, hanem csak leszármazott osztályokon keresztül használható. A név, azonosító szám és cím minden személy alapvető adatai, amelyeket ez az osztály tárol. Az osztály tartalmaz egy absztrakt metódust (`adatokMegjelenitese()`), amely biztosítja, hogy minden leszármazott osztály implementálja a saját adatainak megjelenítését. Ez a megoldás polimorfizmust tesz lehetővé, mert a szülőosztály referenciája keresztül különböző leszármazott osztályok példányai hívhatják meg saját implementációjukat.

2. Hallgató Osztály

A Hallgató osztály a Személy osztályból öröklődik, és kiterjeszti azt olyan hallgató-specifikus tulajdonságokkal, mint a Neptun kód, státusz (aktív/inaktív) és a regisztrált tantárgyak listája. Az osztály felelős a tantárgyak regisztrálásáért, amely során ellenőrzi, hogy a hallgató ne lépje túl a maximális kreditkorlátot (30 kredit). A tantárgyregisztráció csak akkor sikeres, ha a hallgató státusza aktív, és a tantárgy kreditjei beleférnek a még fennmaradó keretbe. A hallgatók vizsgákra is jelentkezhetnek, de csak olyan tantárgyakra, amelyekre már regisztráltak. Az osztály tartalmazza a hallgató adatainak szöveges megjelenítését is, amely a név, Neptun kód, státusz és a regisztrált tantárgyak listáját írja ki.

3. Oktató Osztály

Az Oktató osztály szintén a Személy osztályból származik, és az oktatók egyetemi szerepköreit modellezi. Az osztály tárolja az oktató intézetét, beosztását és az oktatott tantárgyak listáját. Az oktatók hozzárendelhetnek tantárgyakat magukhoz, amelyeket később oktatnak. Az oktató adatainak megjelenítése során a rendszer kiírja az oktató nevét, intézetét, beosztását és az oktatott tantárgyak listáját, így áttekinthetővé téve az oktató tevékenységét. Ez az osztály lehetőséget biztosít arra, hogy az egyetem nyomon kövesse, mely oktatók mely tantárgyakat tartják.

4. Adminisztrátor Osztály

Az Adminisztrátor osztály a Személy osztály egy másik leszármazottja, amely az egyetemi adminisztratív feladatokat látja el. Az osztály tárolja az adminisztrátor jogosultsági szintjét, és lehetőséget biztosít a hallgatók státuszának módosítására (pl. aktívból inaktívvá tétele). Ez a funkció kritikus fontosságú, mivel biztosítja, hogy csak jogosult személyek módosíthassák a hallgatói státuszt. Az adminisztrátor adatainak megjelenítésekor a rendszer kiírja az admin nevét és jogosultságát, így egyértelművé téve, ki milyen szintű hozzáféréssel rendelkezik.

5. Tantárgy Osztály

A Tantárgy osztály az egyetemi kurzusok adatait tárolja, beleértve a tantárgy nevét, kódját, kreditértékét és a rá regisztrált hallgatók listáját. Az osztály lehetővé teszi, hogy hallgatók hozzáadásra kerüljenek a tantárgyhoz, valamint kilistázza a regisztrált hallgatókat. Ez az osztály központi szerepet játszik a tantárgyak nyilvántartásában, és biztosítja, hogy minden tantárgyhoz pontosan nyomon lehessen követni, mely hallgatók vettek részt rajta. A tantárgyak kreditértéke kulcsfontosságú a hallgatói terhelés szabályozásában, ezért ezek az adatok alapvetőek a rendszer működéséhez.

6. Vizsga Osztály

A Vizsga osztály a tantárgyakhoz kapcsolódó vizsgákat kezeli. Minden vizsga társul egy tantárgyhoz, és rendelkezik egy időponttal, valamint a jelentkezett hallgatók listájával. Az osztály lehetővé teszi, hogy hallgatók vizsgára jelentkezzenek, és később az oktatók vagy adminisztrátorok jegyet rögzíthessenek a vizsgán résztvevőknek. A vizsgára csak olyan hallgatók jelentkezhetnek, akik regisztráltak a tantárgyra, és aktív státuszúak. A jegy rögzítésekor a rendszer ellenőrzi, hogy a hallgató valóban jelentkezett-e a vizsgára, így biztosítva az adatok érvényességét. Ez az osztály nélkülözhetetlen a vizsgák szervezésében és eredményeinek nyilvántartásában.

Főbb Funkcionalitások

1. Hallgatói Funkciók

Tantárgy regisztráció:

A hallgató csak akkor regisztrálhat tantárgyat, ha a kreditkorlát ($MAX_KREDIT = 30$) nem éri el a maximumot.

```
if (kreditekSzama + tantargy.getKredit() < MAX_KREDIT) {  
    felvettTargyak.add(tantargy);  
    kreditekSzama += tantargy.getKredit();  
    tantargy.hallgatoHozzaad(this);  
    return true;  
}
```

Vizsgára jelentkezés:

Csak aktív hallgató jelentkezhet, és csak olyan tantárgyra, amire regisztrált.

```
if (aktive && felvettTargyak.contains(vizsga.getTantargy())) {  
    vizsga.hallgatoHozzaad(this);  
}
```

2. Adminisztrátori Funkciók

Hallgató státusz módosítása:

Az adminisztrátor átállíthatja a hallgató státuszát (aktív/inaktív).

```
public void hallgatoStatusztModosit(Hallgato hallgato, boolean ujStatusz) {  
    hallgato.setStatusz(ujStatusz);  
}
```

Vizsga Kezelése

Jegy rögzítése:

Csak a vizsgára jelentkezett hallgató kaphat jegyet.

```
if (jelentkezettHallgatok.contains(hallgato)) {  
    System.out.println(hallgato.getNev() + " vizsgája rögzítve. Jegy: " +  
    jegy);  
}
```

Tesztelés és Példa Futás

1. Tesztesetek

Hallgató létrehozása és tantárgy regisztrálása:

```
Hallgato hallgato1 = new Hallgato("Kiss Péter", "123456", "Budapest", new  
Date(2000-4-20), "ISHW48", true, new ArrayList<>(), 10);
```

```
Tantargy prog1 = new Tantargy("Programozás I", "BMEVIAUAB01", 5);
```

```
oktato.tantargyHozzaad(prog1);
```

Vizsgára jelentkezés:

```
Vizsga prog1Vizsga = new Vizsga(prog1, "2023-12-15 10:00");
```

```
hallgato1.vizsgaJelentkezés(prog1Vizsga);
```

Admin státusz módosítása:

```
Admin admin = new Admin("Admin Erika", "111222", "Pécs", new Date(1989-1-  
5), "Státusz módosítás, jegybeírás");
```

```
admin.hallgatoStatusztModosit(hallgato2, false);
```

Vizsga jegy rögzítése:

```
prog1Vizsga.jegyRogzites(hallgato1, 5);
```