

Interfész feladat (alap osztály és interfészek)

- **Feladat 2.3 - Készítsük el a felsorolt osztályokat és interfészeket**
- **IEladható interfész**
 - Tartalmaz egy csak olvasható *Ár* tulajdonságot
- **IKarbantartandó interfész**
 - Tartalmaz egy logikai értéket visszaadó *KarbantartásSzükséges()* metódust, amely majd visszaadja, hogy szükséges-e a megadott termék karbantartása
 - Tartalmaz egy szöveget visszaadó *Karbantartás()* metódust, aminek a visszatérési értéke lesz a karbantartási esemény leírása
- **Termék osztály**
 - Legyen egy kívülről írható/olvasható *Név* tulajdonsága, amihez az értéket a konstruktorban lehessen megadni
 - Valósítsa meg az *IEladható* interfészt
 - Mivel ezen a szinten még nem ismerjük az egyes leendő termékek árképzésének módját, így legyen a kötelezően megvalósítandó *Ár* tulajdonság absztrakt (ennek megfelelően az egész osztály is)

Interfész feladat (Termék leszármazottak)

- **Cipő osztály**

- Legyen a *Termék* osztály leszármazottja, termék névnek automatikusan „Cipő” szöveget adjon át az ősz konstruktorának
- Tartalmazzon egy *méret* mezőt, amit a konstruktorban lehet megadni
- Valósítsa meg az *Ár* tulajdonság lekérdezését, egy cipő ára legyen a mérettől függő: ha a *méret* > 40 akkor 15000 egyébként pedig 14000

- **Virág osztály**

- A *Termék* leszármazottja, névnek a konstruktorban kapott nevet adja tovább
- Tárolja el a virág *korát*, ebből képezze a szükséges árat is: $1000 + kor * 2$
- Valósítsa meg a *IKarbantartandó* interfészt is az alábbi módon:
 - *KarbantartásSzükséges* : minden harmadik hívásnál adjon vissza igazat (a híváskor növelhetjük a virág korát)
 - *Karbantartás* : visszatérési értéke legyen az „Öntözés” szöveg

- **Hűtő osztály**

- Valósítsa meg az *IKarbantartható* interfészt: a *KarbantartásSzükséges* mindig igazat adjon vissza, a *Karbantartás* leírása legyen a „Tisztítás” szöveg

Interfész feladat (fő modulok)

- **EladóModul osztály**

- Tároljon el tetszőleges számú (a maximumot a konstruktorban lehessen megadni) *IEladható* interfészt megvalósító objektumot
- Legyen egy *ÚjTermékFelvétele(IEladható)* metódusa, amelyik felvesz egy új eladható elemet a fenti tömbbe
- Legyen egy *Legolcsóbb()* metódusa, amelyik visszaadja a legolcsóbb termék referenciáját

- **KarbantartóModul osztály**

- Tároljon el tetszőleges számú (a maximumot a konstruktorban lehessen megadni) *IKarbantartandó* interfészt megvalósító objektumot
- Legyen egy *ÚjKarbantartandóFelvétele(IKarbantartandó)* metódusa, amelyik felvesz egy új karbantartandó elemet a fenti tömbbe
- Legyen egy *MindenKarbantartás()* nevű metódusa, amely végignézi az összes eltárolt karbantartandó elemet, lekérdezi, hogy azok igényelnek-e karbantartást, és ha igen, akkor meghívja azok *Karbantartás()* metódusát. Annak visszatérési értékét írja ki a képernyőre

Interfész feladat (kiterjesztés)

- **IVisszaváltható interfész**
 - Legyen az *IEladható* interfész kiterjesztése
 - Tartalmazzon egy *VisszaváltásiÉrtéke(elteltNapokSzáma)* metódust, amely egy egész számot ad vissza
- **VásárlásiUtalvány osztály**
 - Valósítsa meg az *IEladható* és az *IVisszaváltható* interfészeket is, mindkét esetben 1000-et adjon vissza (ár és visszaváltási érték a napoktól függetlenül)
- **Cipő osztály kiegészítése**
 - Valósítsa meg a *Cipő* osztály is az *IVisszaváltható* interfészt
 - A visszaváltási értéket az ár és az eltelt napok száma alapján számoljuk (lineárisan csökkenjen). Ha még aznap visszaváltják, akkor legyen egyenlő az eredeti árral, 50 nap esetén már legyen 0
- **A fenti osztályok és interfészek implementálását követően hozzon létre minta objektumokat, majd próbálja ki a fenti funkciók működését**