

Feladat – NagyZH – FONTOS INFÓK

- Alkalmazni kell a megoldás során a tanult objektum-orientáltsági elveket.
- A bemeneti fájlokat (**stock-toys.json**, **stock-clothes.json**, **stock-chairs.json**, **shipment-1.json**, **shipment-2.json**) teszteléskor a build könyvtárba kell másolni.
- Feltehetjük, hogy a bemenő/teszt adatok helyesek, érvényesek (kivéve, ha a feladat mást nem állít).
- A megoldás teljes forráskódját egyetlen ZIP fájlba csomagolva kell feltölteni.

A feladat leírása

A *main*-ben és ebben a leírásban szereplő osztályok és metódusok igény szerint **átnevezhetők**, az itteni nevek csak példák.

A feladat egy kisbaba termékekkel foglalkozó áruház árukezelésének megvalósítása. Az áruházban különböző kisbaba termékek (**BabyProduct**) érhetők el, a programnak ezek nyilvántartását kell tudnia kezelnie. Jelenleg háromfajta termékkörrel foglalkoznak: játékok ("**toy**"), ruházat ("**clothes**"), etetőszékek ("**chair**"). A kisbaba termékek esetén nyilvántartjuk az azonosítót (szöveg), a nevét (szöveg), az árat (forintban, egész) és a készletmennyiséget (darab, egész). A játékok esetében ezen kívül még ismerjük, hogy hány éves gyerekeknek szánták azt (egész). A ruházatnál a ruha méretét ismerjük (szöveg). Az etetőszékek esetében még azt tudjuk, hogy a szék magassága állítható-e.

Legyen a **BabyProduct** ősosztályban **<<** operátor, amivel tetszőleges kimeneti folyamra kiírható a termék minden adata (beleértve a gyerekosztályokra jellemző extra adatokat is). A tesztкод ezt használja.

Az árukészletet a **Warehouse** osztály egyetlen példánya tartsa nyilván.

Legyen egy *loadProducts* metódus, ami paraméterben megkapja, hogy milyen fajta termékekről van szó ("**toy**", "**clothes**", "**chair**"), és egy JSON fájl nevét, és a fájlból az adott típusú áruk készletét betölti.

Legyen egy *getProducts* metódus, amely visszaadja a tárolt termékeket **vector<BabyProduct*>**-ként.

Legyen egy másik *getProducts* metódus, ami egy paraméterben a típust várja ("**toy**", "**clothes**", "**chair**"), és akkor csak azokat a típusú termékeket adja vissza **vector<BabyProduct*>**-ként.

Legyen egy sablon *getSpecificProducts* metódus, aminek van egy **T** sablonparamétere (**T** itt csak valamelyik gyerekosztály lehet), továbbá paraméterben egy minimális és maximális árat vár; és a határokat beleértve a két ár közé eső, csak **T** típusú termékeket adja vissza egy **vector<T*>** tárolóban.

Legyen egy **Shipment** osztály, ami egy beérkező szállítmányt tárol. Konstruktornak egy JSON fájl nevét kapja, ahonnan beolvassa a raktárba érkező termékeket azonosító szerint, és a hozzájuk tartozó mennyiségeket. Legyen a **Warehouse** osztályban egy *deliverShipment* metódus, ami hozzáadja a készlethez a szállítmányt.

Legyen a **Warehouse** osztályban egy *makeOrder* metódus, amivel egy rendelést lehet végrehajtani. Paraméterben egy azonosítót, egy darabszámot, és a vevő pénzét kapja meg (ez utóbbit **int&**-ként). Ha

érvényes az azonosító, rendelkezésre áll a raktárban az adott darabszámú termék, és a vevő pénze elégséges a teljesítéshez, akkor csökkentse a raktárkészletet és a vevő pénzét a rendelés szerint. Ha ezek közül bármi nem stimmel, ne módosuljon semmi, de kapjunk egy, a helyzetet elmagyarázó hibaüzenetet.

Legyen a **Warehouse** osztályban egy *saveProducts* metódus, ami két paraméterben várja a termékek típusát ("toy", "clothes", "chair") és egy fájlnévet, ahova JSON formátumban mentse el a raktárkészletet, a betöltéssel megegyező formátumban.

Pontozás

- | | |
|---|----------|
| 1. Termékek osztályszerkezete és tárolása a Warehouse osztályban | (8 pont) |
| 2. Termékek kiírása a << operátorral | (2 pont) |
| 3. <i>loadProducts</i> metódus | (6 pont) |
| 4. <i>getProducts</i> metódus | (2 pont) |
| 5. <i>getProducts</i> metódus (típus szerinti lekérdezés) | (4 pont) |
| 6. <i>getSpecificProducts</i> sablon metódus (szintén típus szerinti lekérdezés) | (4 pont) |
| 7. Shipment osztály és a betöltése | (3 pont) |
| 8. Shipment hozzáadása a Warehouse -hoz a <i>deliverShipment</i> metódussal | (3 pont) |
| 9. <i>makeOrder</i> metódus | (4 pont) |
| 10. <i>saveProducts</i> metódus | (4 pont) |

Összesen: **40 pont.**