Programozói dokumentáció

A funkciók célja

A programban 5 darab funkció van. Az 1. Új alapanyag hozzáadása, 2. Új recept hozzáadása, 3. Recept törlése, 4. Keresés a receptek között, 5. Kilépés funkciók vannak. Ezekből számmal lehet választani (1-5 között) Amennyiben nem számot, hanem sztringet (szavakat) írunk be, akkor kiírja, hogy nem számot adtunk neki, írjunk számot és kiírja, újra miből lehet választani. Amennyiben nagyobb számot választunk (írunk be), akkor kiírja, hogy kisebbet válasszunk, ilyen funkció nincs. Mindegyik választás után megtörténik az adott funkció, majd újra visszatérünk a funkciókhoz, míg ki nem akarunk lépni belőle, amely az ötös funkció. A program minden funkciót konzolról hajt végre és mindig kiírja a pontos teendőt a legtöbb kikötéssel, hogy a felhasználó biztosan értse mit kell tennie.

**A programvezérlését képező osztályok és függvények:**

**Material osztály**

Mivel az italok alapanyagokból állnak, ezért alapanyagok(Material) az első osztályom, ahol van egy hashCode() és egy equals(), amely egy generált függvény nálam egy későbbi osztály contains-éhez ugyanis szükséges hozzá. Ennek az oosztálynak konstruktorában van neve és mennyisége, ezért van egy getName() és egy getQuantity() metódusa, hogy azokat visszaadja meg egy String() metódusa is, ahol formázva kiírja majd, hogy az alapanyag neve és annak a mennyisége ml-ben ha hozzáadjuk az egyes funkcióban. Úgy gondoltam, hogy az italokhoz csak elég tényleges italok, ezért a „ml” és felesleges hozzáadni továbbá olyanokat, mint egy citrom, mert ugyanúgy citromlevet is lehet (akár kifacsarva a citromból).

**Drink osztály**

Azért választottam az italokat, mert ital recepteket kell készíteni. Ez tartalmazza az italokat, amelyeknek nevük van valami hozzávalóik, amelyek arraylistek, hogy mikor kettes funkciónál létrehozza az italokat, hozzá lehessen adni több különböző alapanyagot is. Ennek van hashcode() és equals()-a egy későbbi contains függvényhez ugyanúgy, valamint van getName() getMateriallist()-je, hogy vissszaadja őket. Van egy String() metódusa, hogy a 2 funkció végén kiírja az italt és hozzávalóit, amelyek hozzávannak adva már miután hozzáadtuk.

**Recipebook osztály**

Elsőként létrehoztam egy Arraylistet(továbbiakban és drinkslist) materiallist a két előző osztálynak, amely üres, hogy bele lehessen rakni az alapanyagokat meg az italokat. Ezek mind a Programoperaionshöz kellenek ezen az objektumon vannak meghívva a függvények. Ez az osztály tartalmazza a materialokat és a drinkslisteket és azzal kapcsolatos műveleteket is. pl. hozzáadás, törlés, kereséshez szükséges függvények.

add(Material material)

Ez a függvény megnézi, hogy tartalmazza-e(contains) már a materiallist azt az alapanyagot, amelyet bemenetről olvasunk majd be .

getMateriallist()

Visszaadja a materiallist listáját.

listMat()

Ez kiírja az egész materiallistet a legvégén, hogy látható legyen miket raktunk bele és milyen alapanyagokat tartalmaz alkalmazva az egyes funkció legvégén. Előző függvényen megy végig és visszaadja a hozzáadottakat. Első funkcióhoz szükséges

addDrink(Drink drink)

Hozzáadja a drinkslist-hez az italokat, amennyiben még nem tartalmazza, amelyeket igen azt már nem tudják hozzáadni, kettes funkcióhoz szükséges

listDri()

Ez kiírja az egész drinkslistet a legvégén és az összes hozzáadott drinket.

ArrayList<Drink> getDrinkslist()

Visszaadja a drinkslist listáját.

removeDrink(String drinks)

Ital törlésére alkalmas függvény. Megnézi, hogy tartalmazza-e a drinkslist az italt és ha igen akkor törli azt.

searcherDrink(String drink)

Ha a keresésben bevitt ital megtalálható a drinkslistben, akkor behozza azt az italt.

save(), load()

Fájlkezelésre alkalmas függvények kiíratásra és beolvasásra.

**Programoperations osztály**

Létrehozok egy Recipebook objektumot az elején, hogy az őt tartalmazó függvényeket, hogy megtudjam hívni rajta.

addMater(Scanner newmaterial)

A bemenetről kéri a nevét és a mennyiségét, ha befejezte a bevitelt egy exit szót írásával lehet kilépni. B viszi az adatot a felhasználó név és mennyiség sorrendben szóközzel elválasztva, amennyiben nem ezt teszi, nem fogja hozzáadni és nem történik semmi. Ha mindkettőt leírja, akkor hozzá fogja adni a Recipebook add függvényével és a végén a listMat() függvénnyel kilistázza a hozzáadottakat a legvégén.

**addRecipe(Scanner newdrinke)**

Elsőként bekéri az ital nevét. Leellenőrzi, hogy van-e alapanyag felvéve, ha nincs, akkor kéri, hogy vegyél fel egyet és felhozza a funkciókat, amenyiben ide írja „exit”et ki is lehet lépni alternatívának, amennyiben nem ezt akarta az ember hívni ez után ki is írja, hogy miket lehet helyette. HA van alapanyag, felhozza őket és számmal lehet választani belőle, egymás után kell többet írni tehát pl. 0,1 számút akarjuk, akkor a 0 1 számot kell írni szóközökkel. Ha a felhasználó nem számot ír vagy nagyobb számot ír, mint amennyi alapanyag van (itt ki van emelve a felhasználónak, hogy 0-tól van számozva, de ha nem onnantól számolja nem fog elszállni) akkor lekezeli ezeket a hibákat. Akkor merülhet fel probléma, ha nem rak szóközt, akkor sem hibára fut, csak lehet, hogy rosszul adja hozzá, de ebben az esetben tudja törölni és újra hozzáadja, vagy ugyanazt a nevet beírva jó számokat beírva hozzáadja és későbbiekben törölhetőek a rontott receptek. A **recipebook addDrink** függvényével adja hozzá ezért, ha már tartalmazza azt az italt „Ital már tartalmazza” szöveg jelenik meg a felhasználónak (mennyiségben és névben is megegyezik). A legvégén kilistázza a drinkslistet és a Drink string() metódusa szerint jelennek meg az alapanyagok hozzá. A legvégén pedig kiírja fájlba az italokat a save() metódussal. Azért gondoltam így ezt a függvényt, hogy leellenőrizze a hibákat valamint a felhasználónak könnyebb legyen, és pontosan le van írva, hogy mit kell tennie a kimeneten. Ha mégsem így történik, akkor se legyen nehéz orvosolni a problémákat. A kiíratás azért történik, hogy ha újra indítjuk a programot, kilépés után az eddigiek ne szálljanak el. A fájlkezeléses hibát is lekezeli.

**removeRecipe(Scanner newdrinke)**

Itt megkérdezi, hogy melyik receptet szeretnénk törölni és a neve bevitelével törölhetőek ezek. Betölti az eddig hozzáadott recepteket, hogy az előzőeket is lehessen törölni. Ha megegyezik a neve az italoknak, akkor a sorban az elsőt fogja törölni. A receptkönyv törlési függvényével meghívva törli és kiírja, hogy ital törölve, ha tudja törölni, ha nem, akkor nincs mit törölnie és egyből a recept listát hozza be, amelyek törölhetőek. Ha a szóköz a végén beesik, akkor csak úgy fogja törölni, ha azt is mögé írja a felhasználó, ha nem, akkor nem csinál semmit. Erről tájékoztatva van a felhasználó is a kimeneten. Ha nincs mit törölni, kiírja, hogy „Nincsenek italok”. El is menti a végén és kiírja fájlba a törléseket. A fájlkezeléses hibát lekezeli.

**searchRecipe(Scanner searching)**

Ebben a függvényben kiírja, hogy milyen italnevek vannak, melyik receptet lehet megnézni és ennek nevét beírva a bemenetbe fogja megjeleníteni annak az italnak a hozzávalóit. Azért gondoltam, hogy lehívom milyen italnevek vannak, hogy a felhasználó tudja, hogy miből lehet választani és tudja melyik nevű receptre lehet kíváncsi. Ha nincsenek italok, akkor kiírja és megkér, hogy vegyél fel és hol tudsz. Ez a keresés a Recipebook objektum searcherDrink függvényével fut(kikeresi a drinkslistből a megfelelő italt és majd visszaadja azt). Betölti itt is a fájlból milyen italok vannak jelenleg, ezt lekezeli, ha nincs ilyen fájl.

**Main osztály**

Ez az osztály a futásért felelős egyből behozza a funkciókat, amelyet a dokumentáció elején már ismertettem. Lekezeli, ha nem számot ír be a felhasználó, ha pedig nagyobb számot ír be megkér, hogy kisebb számot írjon be, 5. számmal ki lehet lépni. Ha kiválasztottuk, akkor a program a Programoperations megfelelő függvényét meghívja, amelyek a Recipebook objektumokon futnak és végrehajtódik a kért menüpont.

**Felhasználó kézi könyv**

A főprogram felületéről számokkal lehet választani 1-5 között. Amennyiben hozzá szeretne adni, akkor válassza a 1 számot, ha italt szeretne hozzáadni 2 számot, ha italt törölni a 3 számot, ha italt keresne a 4 számot, amennyiben kilépne a programból, az 5 számot kell választania. Az első pontban meg kell adni a hozzávaló nevét és mennyiségét (ez egy szám lehet). Csak itallevet lehet hozzáadni, amennyiben ki akarná egészíteni citrom vagy egyéb gyümölcstermékkel, akkor annak a levét kell hozzáadnia (citromlé, narancslé stb. ebben az esetben egységet maga választ, de lehet az 1) is. A kettes pontban először az ital nevét kell megadni (figyeljen a szóközökre!) majd enter nyomásával megjelennek az alapanyagok, amelyek választhatóak. Ha még nincs alapanyag hozzáadva kérem először ezt tegye meg. Ha van akkor számokkal választhatóak melyik alapanyag szükséges (0-val kezdődik az első alapanyag számozása). Ezeket egyszerre kell bevinni enterrel elválasztva. Ha mégis elrontja, akkor nyomjon entert, majd válassza a hármas funkciót és adja meg az ital nevét, amelyet törölhet. Ezután újra visszatérhet a hozzáadáshoz és helyesen megadhatja az adatokat. Amennyiben csak törölni szeretne egy italt, amelyet már nem szeretne, írja be a nevét, majd megjelenik a továbbiakban törölhető elemek, ez akkor is megjelenik, ha elrontja az ital nevét. Amennyiben egy szóközt tett a végére nem fogja törölni, csak ha ugyanazt a szóközt, hozzáteszi és a név teljesen megegyezik. Amennyiben keresni szeretne az a négyes számmal érhető el. Ezt megnyomva megjelennek, hogy milyen italok vannak a recepttárban és tekinthetőek meg azok alapanyagai. Az ötös gombbal pedig kiléphet abból, majd újra visszaléphet. Az elmentett italok és hozzávalók nem fognak elveszni.