

Feladat

Egy állatkereskedésben különféle állatokat (hörcsög, pinty, tarantulla, ...) árulnak. Egy állatot egyedi sztring azonosít, ismert a színe, az értéke, de az aktuális árát úgy számoljuk ki, hogy az értékét az állat jellemzőjétől (pl. fiatal vagy felnőtt) függő szorzótényezővel korrigáljuk. Az állatkereskedések nyilvántartják a partnereiket: akiktől beszerezik az állatokat, illetve akiknek eladják azokat. Mind a beszerzéseket, mind az eladásokat számlák igazolják. Egy számla (ami tehát lehet beszerzési vagy eladási) tartalmazza, hogy melyik kereskedés melyik partnerrel melyik kisállatra mikor és mennyiért kötött üzletet. Ugyanazon állat ára szerződésenként eltérő lehet, de ezek alapján kiszámolható egy átlagár.

Szorzó	fiatal	felnőtt
hörcsög	2.0	1.0
pinty	1.0	3.0
tarantulla	3.0	2.0

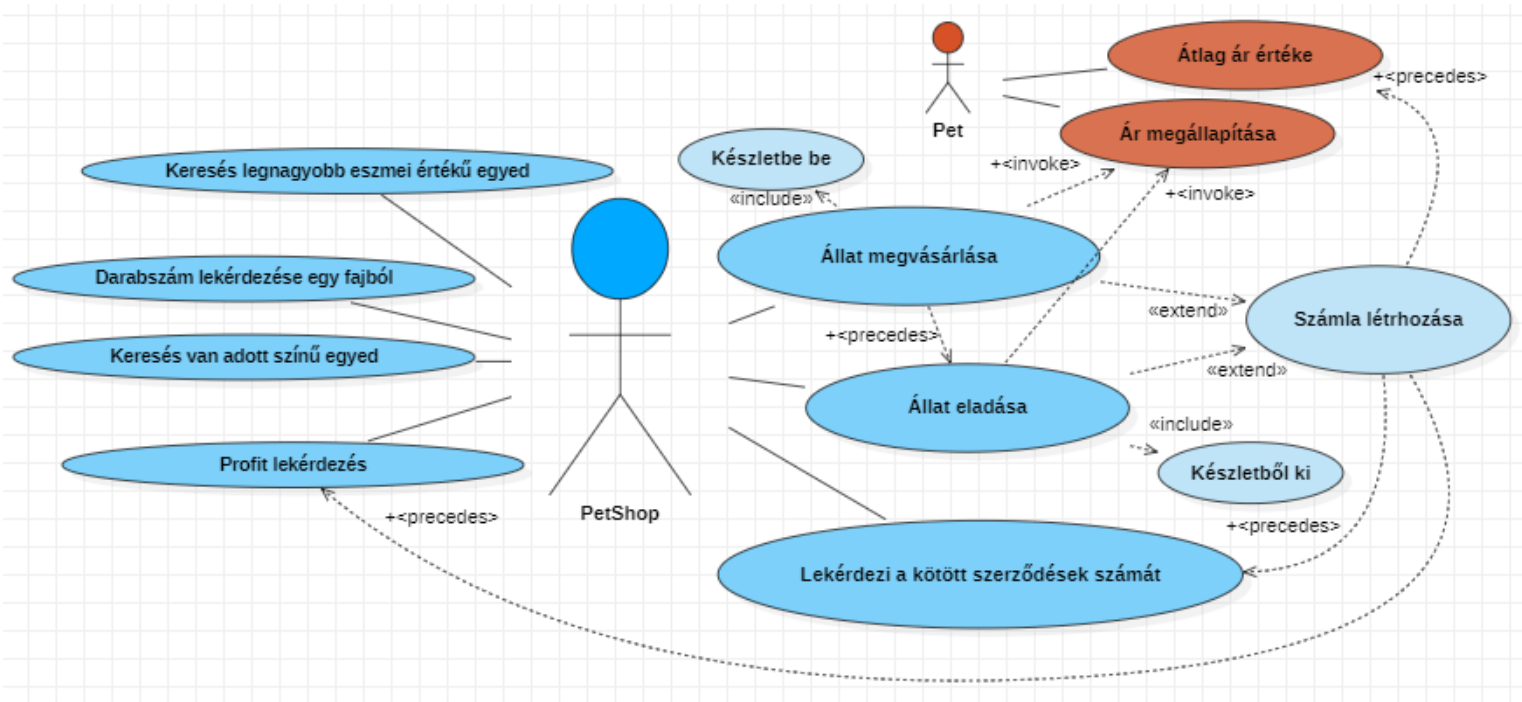
Kérdések

- Van-e egy kereskedésben adott színű pinty?
- Hány hörcsöge van egy kereskedésnek?
- Melyik a legnagyobb eszmei értékű tarantullája egy kereskedésnek?
- Hány szerződést kötött egy adott kereskedés egy adott partnerrel?
- Mekkora egy kereskedésnek a nyeresége (eladási számláin szereplő árak összege mínusz a beszerzési számláin szereplő árak összege)?

Terv

Használati eset diagram

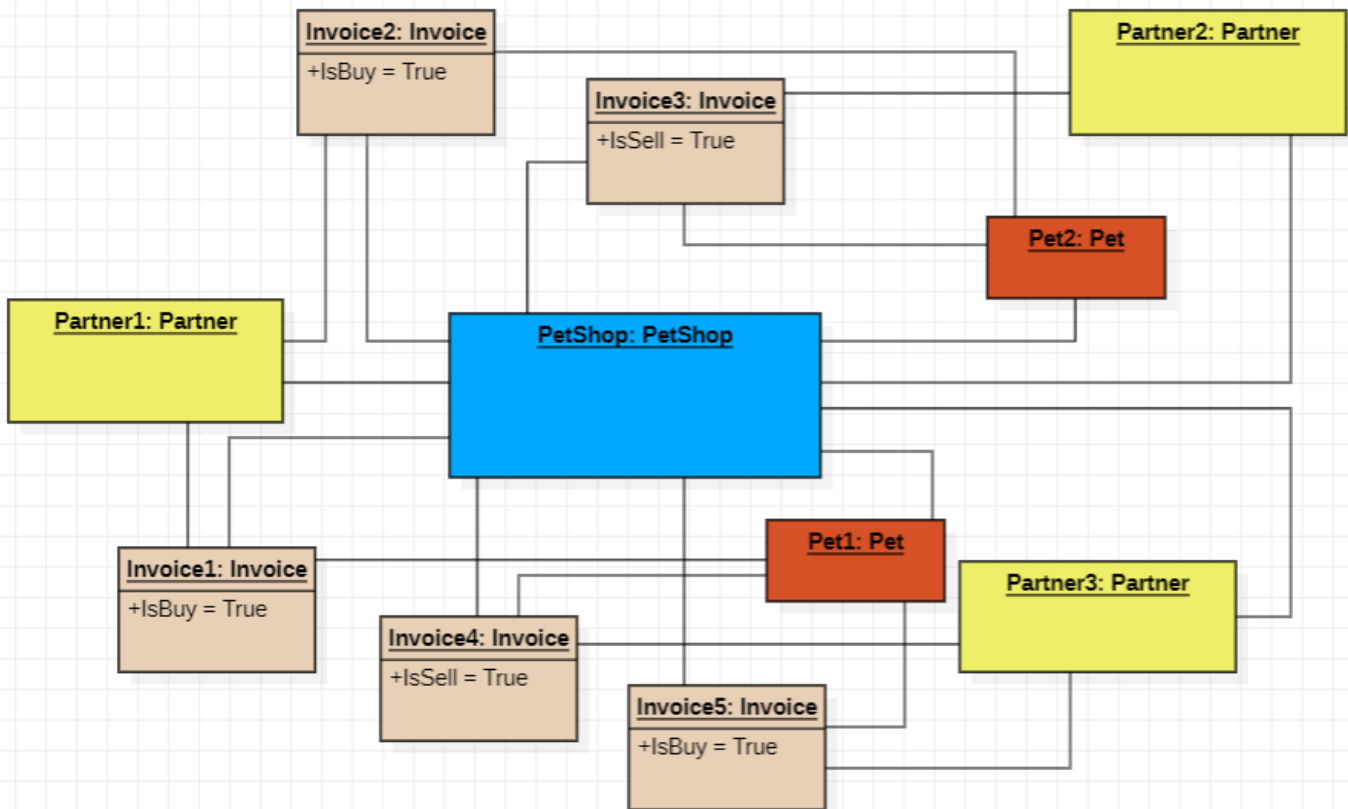
A feladat központi eleme a kisállatokkal való kereskedés és a hozzájuk tartozó egyéb eseményke. Egy háziállat megvásárlásakor keletkezik egy számla illetve a kereskedés tulajdona lesz. Az állat eladásakor ismét számla keletkezik és ekkor kikerül a kereskedés tulajdonából az egyed. A kereskedés ezenkívül még információt gyűjthet a készletéről illetve a nyereségéről. Ezenkívül feltüntettem azon metódusokat is amelyek az egyes részfeladatoknál lesznek alkalmazva.



A vásárlás és eladás hasonlóan működik és hívásaik is megegyeznek csak irányuk más(a kiskereskedés szemszögéből). A metódusok meghívásakor változik a készlet állapota, ezzel együtt az adott egyed ára is megállapításra kerül végül ha minden adat rendelkezésre áll kiállításra kerül a számla. Elérhető az egyed átlag ára is ami a már róla létező számlák alapján kerül meghatározásra. A PetShop pedig a rendelkezésre álló számlák segítségével számolja ki a profitot illetve az is meghatározható, hogy egyes partnereivel hányszor üzletelt.

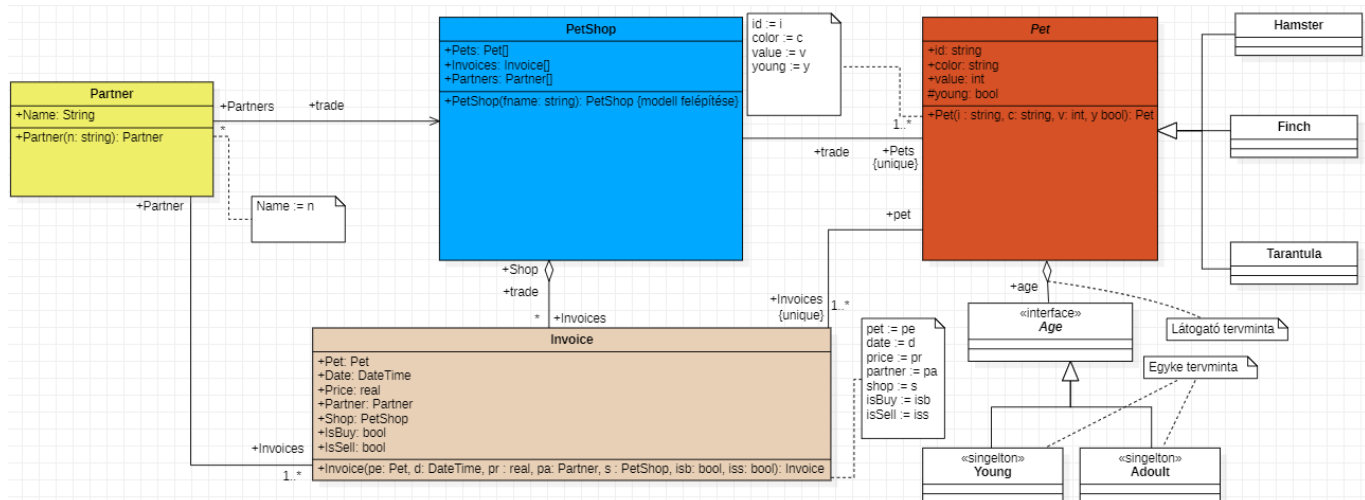
Objektum diagram

A működés során modellben lévő objektumok kapcsolata, kommunikációja az objektum diagramon látható. A fenti feladathoz ez egy olyan objektum diagram, amely mutat egy kereskedést, három partnert, két állatot, és néhány szerződést; az állatok közül az egyik éppen a kereskedés tulajdona (eggyel több vételi szerződése van rá, mint eladási).



Osztály diagram

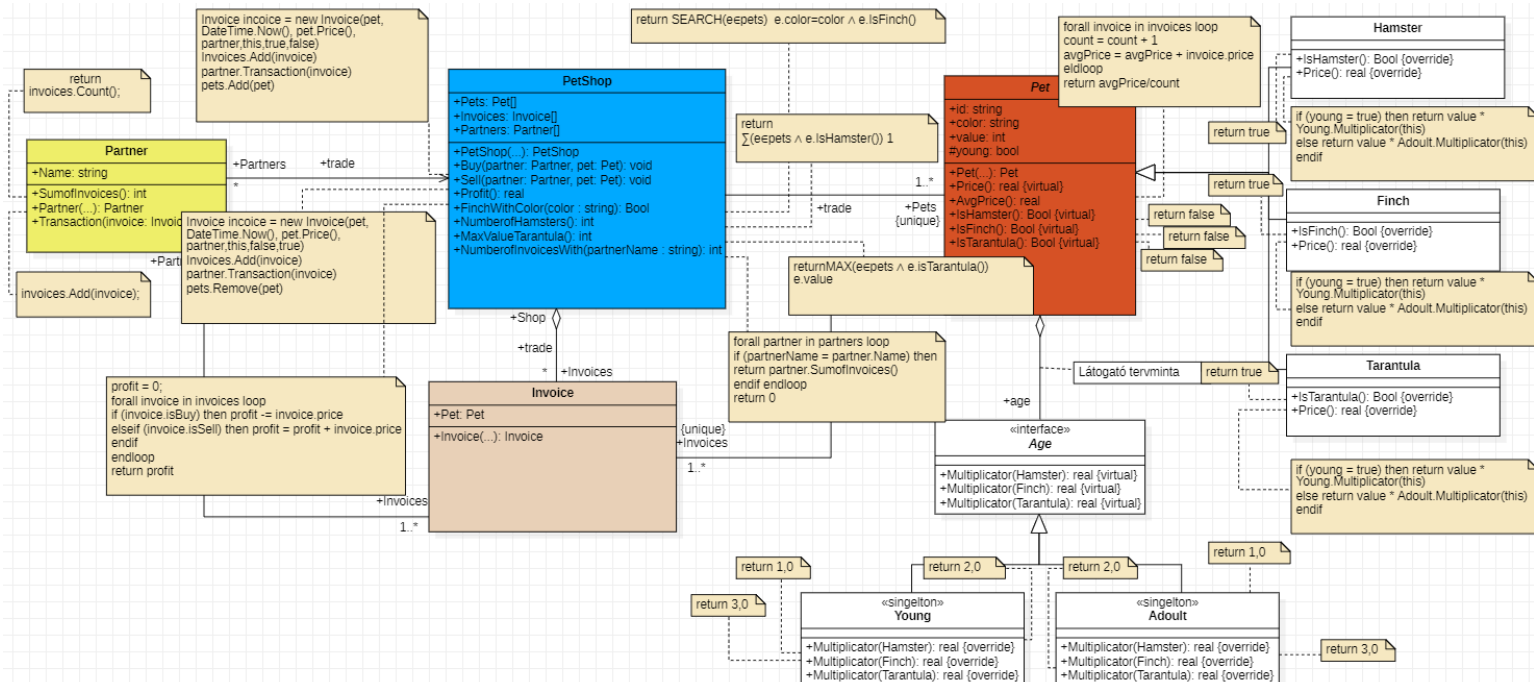
A modell felépüléséhez tartozó osztály diagram kapcsolatokkal és konstruktoraikkal.



A különféle állatfajtákat a Pet osztályból való származtatással adjuk meg. Illetve az Age osztályból is származtatással hozzuk létre a fiatal és felnőtt egyedek szorzó értékét tartalmazó osztályokat. Az osztályok közötti kapcsolatok trade, azaz az eladások és vásárlások lebonyolításakor jönnek létre. Ez a két metódus(Buy()/Sell()) felel PetShop osztály listáinak folyamatos frissítéséért és a számlák kiállításáért is.

Kiegészített osztály diagram

Az osztálydiagram az objektum-kapcsolatokat létrehozó metódusokkal, valamint a feladat kérdéseit megválaszoló metódusokkal való kiegészítése.

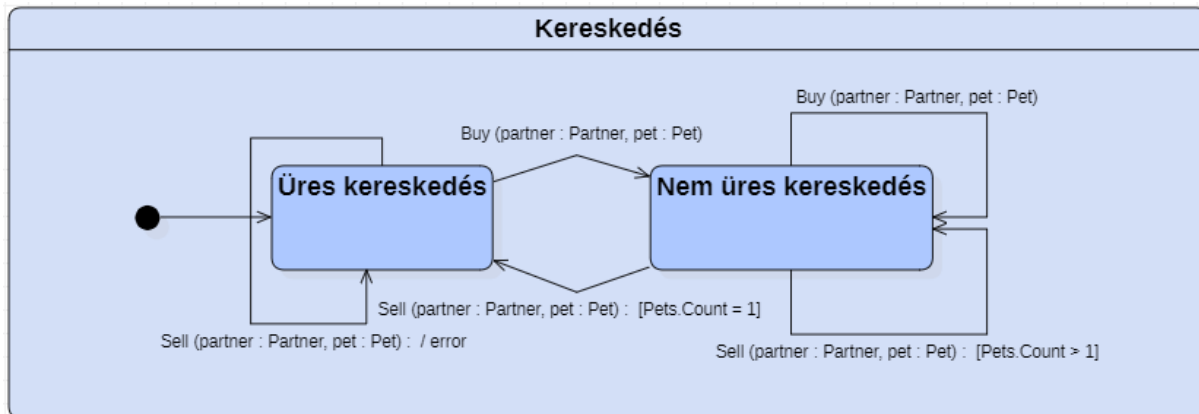


Mivel a Price metódus egyszerre függ az adott állat fajtától és a korától, ezért alkalmazzuk a látogató tervezési mintát: egy állatnak egy kor objektum a „látogatója”. Minden alosztály annyi Multiplier() metódust tartalmaz, ahányféle állat van. Illetve a kor objektumok nem tartalmaznak adattagokat, így egy korokból elég egyetlen objektumot példányosítani: legyenek a Age osztályai az „egyke” tervminta megfelelőek.

A feladatok megoldásához alapvető algoritmusokat használtam, keresés, maximum kiválasztás illetve különféle megszámlálásokat, összegzéseket. A metódusok populálás során kialakított kapcsolatokat, listákat használják és azok lekérdezésével adnak eredményt.

Kereskedés állapotgépe:

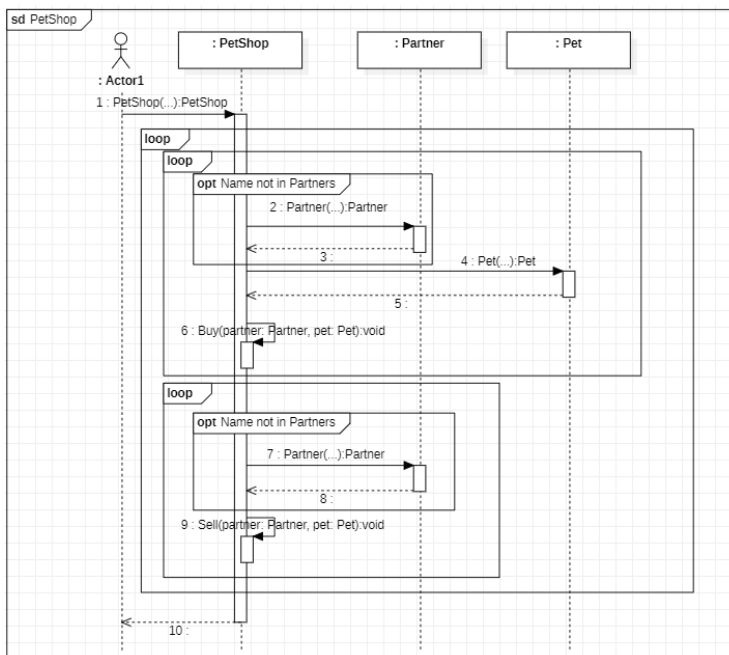
Egy kereskedés állapotát a kereskedésben árult állatok gyűjteménye határozza meg: ennek megfelelően a kereskedés lehet üres vagy nem üres azaz két állapota van. Az állapot-átmeneteket megvalósító tevékenységeket a PetShop osztály metódusaiként definiáltam.



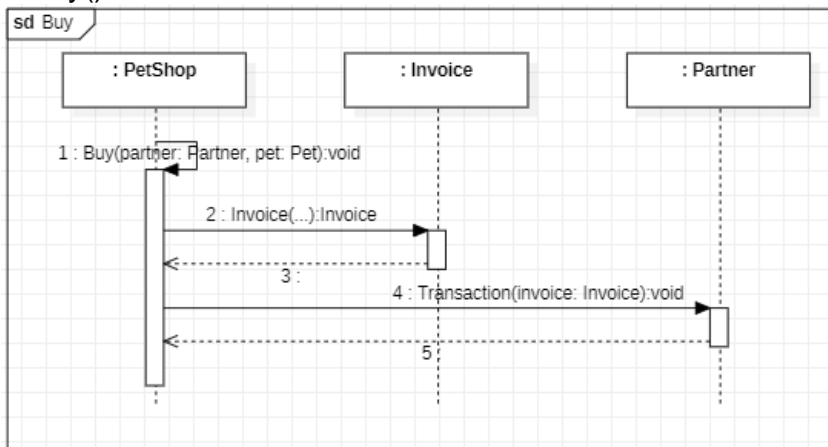
Az állapotok közötti mozgást a **Buy()** és a **Sell()** metódus teszi lehetővé ugyanis ezek szolgálnak az egyes egyedek megvásárolásáért – ezzel együtt a PetShop osztály **pets** készlet, állomány listájába tevéséért vagy eltávolításáért. Ahogy a diagramon is látható ha a kereskedés üres nem képes eladni(**Sell()** metódus), azonban ha vásárol(**Buy()** metódus) egy egyedet akkor Nem üres állapotba vált, azután mind addig amíg a **Buy()** metódussal vásárol vagy a **Sell()** metódussal elad de még több mint 1 állata megőrzi az állapotát. Abban az esetben ha pillanatnyi egyed száma 1 akkor **Sell()** estén állapotot vált és ismét Üres kereskedés lesz.

Szekvencia diagrammok

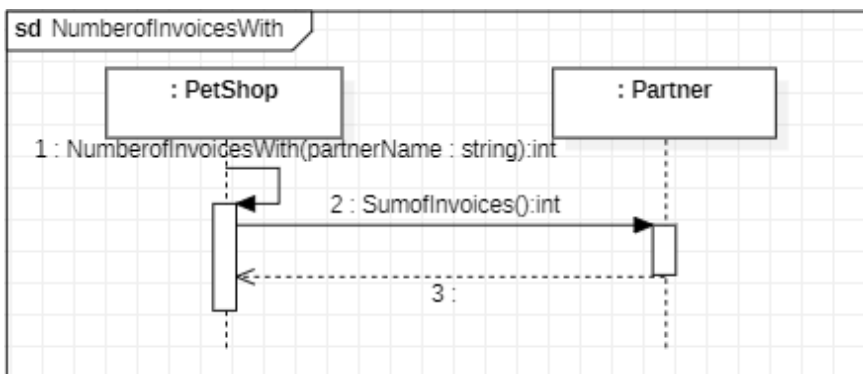
A diagram a PetShop osztály konstruktorának meghívását mutatja be amikor felpopuláljuk a kereskedést.



A Buy() illetve a Sell() metódusok a függvényhívásaik tekintetében, azonosnak tekinthető alább a Buy() metóduson keresztül van bemutatva.



Még egy példa az egyes partnerekkel kötött szerződések számát meg adó lekérdező függvény működése:



Tesztelési terv

A teszteseteknek számos esetet le kell fedniük, hogy teljeskörűen lefedjék az egyes programok működését és függvényhívásaik útvonalat. Az egyes függvényekhez különféle eseteket szükséges meghatározni.

A tesztesetek tartalma

Petshop metódusai – ezen metódusok tesztelésével szinte az egész program letesztelhető ugyan is az egyes osztályok metódusait is meghívják az eredmény adáshoz. Azaz a többi osztály metódusait a PetsShop osztály tesztesetein keresztül azokat kiegészítve fedjük le.

PetShop populálás – ezekkel az esetekkel a Buy() és Sell(), illetve Profit metódusok is tesztelhetők

1. Nincs eladás, nincs vétel
2. Csak eladás
3. Csak vétel
4. Eladás és vétel is van

FinchWithColor() – Létrehozódnak a Pet/Finch objektumok megfelelőek-e az adattagjaik

1. Nincs pinty a kereskedésben
2. Nincs megfelelő színű pinty
3. Volt de már el lett adva(nem kéne már a listában szerepelnie – Sell() metódus működése megfelelő-e)
4. Van megfelelő színű pinty

NumberOfHamsters() – Létrehozódnak a Pet/Hamster objektumok megfelelőek-e az adattagjaik

1. Nem vásárolt a Kereskedés hörcsögöt
2. Nincs a készletben pillanatnyilag hörcsög
3. Van a készletben pillanatnyilag pontosan 1 hörcsög
4. Van a készletben pillanatnyilag több mint hörcsög

MaxValueTarantula() – Létrehozódnak a Pet/Tarantula objektumok megfelelőek-e az adattagjaik

1. Nincs tarantulla a kereskedésben
2. Van 1 tarantulla a kereskedésben
3. Van több tarantulla a kereskedésben
4. Van több tarantulla a kereskedésben és van olyan akiknek azonos az eszmei értéke

NumberOfInvoicesWith()

1. Nem létező partner
2. Létező partner