

Programozás III - Vizuális és Eseményvezérelt Programozás (C#)

Téma: **Első zárthelyi dolgozat**, Alapvető vezérlők

Tagozat / évfolyam: **nappali**, II. évfolyam

Dátum: **2015.04.02.** (április, csütörtök)

Kidolgozási idő: **100 perc**

Oktató: **Bedők Dávid** <http://www.qwaevisz.hu>

Neptun kód: _____ Aláírás/Név:

A feladat megoldása során csak offline súgót és az IntelliSense-t lehet használni. A feladatot óra végén hálózaton keresztül kell majd beadni. Csak az a feladat kerül értékelésre, mely hiba nélkül lefordul. Az alkalmazás értékelésében a nem OOP szemléletű megoldásért hibapont jár, illetve a GUI felületének hanyag kivitelezése is okozhat pontlevonást.

Készítsen Visual Studio 2013-ban ablakos alkalmazást (WPF), mely az alább leírt feladatra ad megoldást. Ahol jelezve van, *adatkötést* használjon.

Mobiltelefon szerződések szeretnénk karbantartani egy ablakos alkalmazás segítségével. Vannak *kártyás* (prepaid) és *előfizetéses* (postpaid) csomagok, és bár minden szerződésnek vannak egységes adatai (pl. szerződő neve), a kezelésük alapvetően különböző felületekkel oldható csak meg. A megrendelő keresni és listázni szeretné a rendszerben lévő szerződéseket egy adott időszakra nézett összköltség alapján.

A kártyás szerződésekről az alábbi adatokat szeretnénk eltárolni: **szerződő neve**, **életkora**, választott **csomagja** (felsorolás: Basic (50), Simple(20), Student(30)) és kiemelt **időszakja** (felsorolás: Morning, Afternoon, Evening, Weekend). Az előre fizetett csomagok díjszabása nagyon egyszerű: a csomag egységárának (zárójelben lévő érték) megfelelő értéket fizet SMS-ért és percenkénti hívásokért minden időszakban, kivéve a kiemelt időszakot, amikor ennek a felét. Az előfizetéses szerződésekről az alábbi adatokat szeretnénk eltárolni: **szerződő neve**, **életkora**, **percdíja** és **SMS költsége**, illetve hogy kér-e a csomagjához **3G/LTE lehetőséget** (logikai érték, kér/nem kér). Az előfizetéses szerződések jelen modellben teljesen egyéniék (nincsenek csomagok).

Minden szerződéshez tartozik egy **tranzakciós lista**, pl. az alábbiak szerint: 5 perc telefonálás reggel, sms küldés, 2 perc telefonálás este, 2 sms küldés, stb. (ez egyfajta időrendi lista egy adott időszakra nézve).

Definiáljon legalább 5 kártyás és 5 előfizetéses szerződést, és rögzítsen mindegyikhez 5-10 véletlenszerű tranzakciót is! (a modellben minimum 4 osztályt kell meglátni)

Készítsen egy űrlapot, melyen egy listában látszódjon az összes szerződés (alapadatok jelenjenek meg (ToString())). Legyen egy **minimum** és **maximum** mezőben definiálható az aktuális időszakra nézett összesített tranzakciós költség, és a **Keresés** gombra való kattintás hatására a szűrési feltételnek megfelelő szerződések töltődjenek be egy másik listába!

Minimum: 0	Maximum: 1000	Keresés
John S., PrePaid...	Melinda K., PostPaid...	
Melinda K., PostPaid...	Peter P., PrePaid...	
Peter P., PrePaid...		
Daren J., PostPaid...		

Bármely listán (!) dupla kattintás hatására egy új modálisan megnyíló ablakban jelenjen meg a szerződés minden adata (alapadatok, tranzakciós adatok költséggel együtt, összesített tranzakciós költség). Különbőféle modális ablak jelenjen meg annak függvényében, hogy milyen szerződést választott ki (vagyis két új ablakot kell készíteni).

A modális ablakoknak nem szükséges, hogy közös ősök legyen, de a szerződések modelljében jelenjen meg a közös ő. A szerződés típusát az osztály dinamikus típusa hordozza!

A modális ablakon megjelenő vezérlők értékei adatkötés segítségével legyenek feltöltve. A logikai értéket (3G/LTE) jelölő négyzettel vizualizálja.