

# ProgramTECH adatátvétel számlázáshoz

ProgramTECH Számlázó Program és Házipénztár Program 2-es csomag  
ProgramTECH MOST Ügyviteli Szoftver Számlázó Modul és Hálózati Modul

Dokumentáció változatszám: 2.6

Frissítve 2024.02.02.

## Tartalom

<a href="#">Bevezető.....</a>	<a href="#">1</a>
<a href="#">Adatátvétel Windows alkalmazásból.....</a>	<a href="#">2</a>
<a href="#">Adatátvétel webes alkalmazásból.....</a>	<a href="#">2</a>
<a href="#">Adatforrás formátuma.....</a>	<a href="#">3</a>
<a href="#">Átadható adatmezők és típus.....</a>	<a href="#">3</a>
<a href="#">Megvalósítás.....</a>	<a href="#">7</a>

### Bevezető

Mindkét fent megnevezett szoftverünk esetében az adatátadás azonos módon történik. A ProgramTECH MOST mivel komplexebb rendszer, így több paraméter átvételére képes.

Az adatátadás során minden bizonylattal kapcsolatos paraméter átadható, amit a bizonylat elkészítése során a számlázó program egyébként is támogat. Érvénytelenítő, helyesbítő, előlegrszámla és ennek végszámlája adatátadással nem, hanem közvetlenül a számlázó programból készíthető el.

Az illesztéshez szükséges átadó, illetve weboldali feltételeket teljesítő modulok és beállítások nem részei a ProgramTECH Számlázó Program és Házipénztár Programnak vagy a ProgramTECH MOST-nak. Ezek elkészítéséről a Felhasználónak kell gondoskodnia. Igény esetén külön megállapodás keretében vállaljuk azok elkészítését. Ez a dokumentáció minden szükséges információt tartalmaz ahhoz, hogy a Felhasználók maguk vagy az általuk választott szakember segítségével az illesztést elkészítsék.

### Adatátvétel Windows alkalmazásból

Abban az esetben ha Önnek van már egy működő Windows alapú ügyviteli rendszere, de az nem rendelkezik számlázó modullal, akkor viszonylag egy egyszerű illesztő program elkészítésével megoldható, hogy az Ön rendszerében létrejött adatok alapján egy bizonylat elkészüljön.

Az Ön szoftverében kell készíteni egy modult amely a lenti specifikációnak megfelelő formátumú és adattartalmú fájlt hozza létre. Ezután a számlázó programot meg kell hívni és paraméterként az adatokat tartalmazó fájl nevét kell megadni. Például:

```
C:\elérésiút\szamlazo.exe c:\temp\szamla.txt
```

A fenti parancs hatására elindul a számlázó program. Azonnal ellenőrzi, hogy létezik e a megadott fájl, illetve annak tartalma megfelel e a lenti specifikációnak, azaz szerepel e minden kötelező adat.

Ha az adatok beolvasása megtörtént akkor a fájl kiterjesztését a számlázó program kiegészíti \_1-el, tehát `szamla.txt_1` lesz a fájl neve.

Ha a nyomtatás is elindult, akkor a fájl nevét a számlázó program átnevezi `szamla.txt_2`-re. Ez azt jelenti, hogy a számlázó program elkezdte a bizonylat nyomtatását, sőt el is végezte, de nem garantálja azt, hogy a nyomtatás valóban megtörtént.

Ennek segítségével a hívó programból is megállapítható, hogy a bizonylat valóban elkészült vagy sem.

### Adatátvétel webes alkalmazásból

Abban az esetben, ha Önnek van egy működő webes alkalmazása (jellemzően webáruház), akkor könnyen megoldható az, hogy a webes alkalmazás adatbázisából (például MySQL) egy azonosító segítségével a bizonylat elkészítéséhez szükséges adatok elérhetőek legyenek.

A számlázó program beállítható URL-en keresi azt a fájlt ami a szükséges kimenetet biztosítja. Például: `https://www.webboltom.hu/szamlazok.php?azon=12345`

Bizonylat készítésekor rá kell kattintani az új ügyfél felvitele gombra, majd a webes számlázás gombra kell kattintani. Ott be kell írni az azonosítót, amely lehet bármilyen adat, a lényeg, hogy a fenti php program azt kezelni tudja, illetve egyértelmű legyen, hogy mit is kell a számlára tenni.

### Adatforrás formátuma

Az adatforrás formátuma fájl és webes illesztés esetében is teljesen azonos. Webes illesztés esetében a böngészőben ugyanúgy kell látni az adatokat mint ahogy azt egy fájl megnyitásakor látnánk.

A forrás adatok UTF8 karakter kódolásban érkezhettek.

Minden adat egy sorban szerepel, a sorvégén chr(10) és chr(13) karakterek. A sor elején [ ] jelek között szerepel a sorban található adatmező neve. Az adatmezők bármilyen sorrendben lehetnek.

A megadott hossznál nagyobb karakterek kérdés nélkül lemaradnak.

A feldolgozás a fájl végéig tart, üres sorok lehetnek. Példa egy adatsorra:

```
[VEVOEMAIL]pelda@valami.hu
```

A bizonylat keltje mindig az adatforrás beolvasásának a PC-n beállított óra dátuma, de ez bizonylat nyomtatása előtt állítható. Ha van már számla az adott naphoz képest a jövőben akkor nem készül el a számla.

Az ügyfél és tétel adatokat minden esetben át kell adni. Az ügyfél esetében az egyedi kulcs az ügyfél neve, a tételek esetében a tétel neve. Tehát minden ügyfélnévnek és tételnevnek egyedinek kell lennie. Ha az ügyfél kapott már számlát, akkor is minden adatát át kell adni.

A \_ (aláhúzás) karakterrel megadott mezőnevek kötelezőek.

### Átadható adatmezők és típus

Mező neve	Típus	Leírás
[FILE_ELEJE]		Az adatokat tartalmazó fájl vagy html kimenet első sorában ennek kell lennie.
[BIZONYLAT_TIPUSA]	Char	Az alapértelmezett érték SZÁMLA. Lehetséges értékek: SZÁLLÍTÓLEVÉL, SZÁMLA, MEGRENDELÉS vagy DÍJBEKÉRŐ.
[TOMB_ELNEVEZESE]	Char(30)	*MOST Abban az esetben, ha több tömb is létezik az adott bizonylattípuson belül, akkor szükséges a tömb elnevezésének átadása is. Például: HUF Számlák.
[_VEVONEVE]	char(100)	Vevő neve.
[UGYFELKOD]	char(30)	*MOST Ügyfél egyedi kódja. Ha meg van adva, akkor az azonosítás végig az ügyfélkód szerint történik.
[VEVONEVE_EGYEB]	Char(100)	Egyéb adat, mely a vevő neve alatt lesz a számlán kinyomtatva.
[VEVOORSZAG]	Char(30)	Alapértelmezett érték Magyarország. A számlán csak az ettől eltérő országok lesznek kinyomtatva.
[_VEVOIRSZAM]	Char(15)	Vevő címének irányítószáma.
[_VEVOTELEPULES]	Char(50)	Vevő címének településneve.
[_VEVOCIM]	char(50)	Vevő címe (közterület neve,

Mező neve	Típus	Leírás
		hátszám).
[VEVOADOSZAM]	chr(30)	Vevő belföldi adószáma.
[VEVOADOSZAM_KOZOSSEGI]	chr(30)	Vevő közösségi adószáma.
[VEVOADOSZAM_CSOPORTAZO NOSITO]	chr(30)	Vevő csoportazonosító adószáma.
[VEVOEMAIL]	char(60)	Vevő e-mail címe.
[VEVOFAX]	char(20)	Vevő FAX száma.
[VEVOTEL]	char(20)	Vevő telefonszáma.
[VEVOKAPCSOLATTARTO]	char(50)	Vevő kapcsolattartó neve.
[VEVOPOSTANEV]	char(100)	Vevő postázási neve.
[VEVOPOSTAIRSZAM]	char(15)	Vevő postázási irányítószáma.
[VEVOPOSTATELEPULES]	char(50)	Vevő postázási településneve.
[VEVOPOSTACIM]	char(50)	Vevő postázási címe (közterület, hátszám).
[VEVOBANKSZAMLASZAM]	char(32)	Vevő bankszámlaszáma.
[VEVOPOSTAORSZAG]	char(30)	Alapértelmezett érték Magyarország.
[VEVOBANKNEVECIME]	char(50)	Vevő bankjának címe
[VEVOBANKSWIFT]	char(50)	Vevő SWIFT kódja
[VEVOBANKIBAN]	char(50)	Vevő IBAN kódja
[_FIZETESIMODNEVE]	char(15)	Fizetési mód neve
[SZALLITASIHATARIDO]	yyyy.mm.dd	Szállítási határidő vagy napja, értelmezés szerint. Megrendelés esetében a rendelt kiszállítás napja.
[_FIZETESIHATARIDO]	Yyyy.mm.dd	Fizetési határidő napja
[_PENZNEM]	char(3)	Pénznem (bármilyen lehet)
[BANK_SZAMLASZAM] <sup>1</sup>	char(50)	Bizonylat kibocsátó pénznemhez tartozó bankszámlaszáma.
[BANK_NEVE]	char(50)	Bizonylat kibocsátó pénznemhez tartozó bank neve.
[BANK_IBAN]	char(50)	Bizonylat kibocsátó pénznemhez tartozó IBAN.
[BANK_SWIFT]	char(50)	Bizonylat kibocsátó pénznemhez tartozó SWIFT.
[NYOMTATAS_AZONNAL]	char(4)	A nyomtatás az adatok sikeres beolvasása után azonnal elinduljon vagy sem. Lehetséges értékek IGEN és

<sup>1</sup> Ha a számlán fel kell tüntetni bankszámlaszámot, akkor minden alkalommal át kell adni.

Mező neve	Típus	Leírás
		NEM, alapértelmezett a NEM.
[ NYELV ]	char(3)	Lehetséges értékek: HUN, ENG vagy GER. Ha nincs átadva, akkor magyar nyelvű lesz a bizonylat.
[ NYELV2 ]	Char(3)	Bizonylat másodlagos nyelve. Megadása esetén a [NYELV] paraméter lesz az elsődleges, a [NYELV2] pedig a másodlagos. A másodlagos nyelv megadása esetén az elsődleges nyelv megadása kötelező.
[ TELJESITES_DATUMA ]	Yyyy.mm.dd	Bizonylat teljesítés dátuma. Ha nincs megadva akkor a bizonylat keltjével azonos nap.
[ RAKTAR_NEVE_KIADASHOZ ]	char(30)	*MOST Az átadott tétel ebből a raktárból kerül levonásra. Egyébként az alapértelmezett raktárat használja a program.
[ FEJLEC_SZOVEG ]	Szöveg (bármilyen hosszú)	Ez a szöveg a számla fejlécében, számla tárgyaként kerül kinyomtatásra.
[ ESZAMLA ]	char(4)	*MOST Elektronikus számlázás esetén használandó, ebben az esetben értéke IGEN.
[ TERMEKEK_ELEJE ]		Ezután a sor után következhetnek a bizonylat tételei.
[ TERMEK ]		Ehhez nem tartozik adat, itt kezdődik egy tételsor részletezése. Ahány termék, annyi ilyen sor és alatta a részletek.
[ _MEGNEVEZES ]	char(100)	Termék vagy szolgáltatás megnevezése. Ez alapján történik a tétel azonosítása.
[ MEGJEGYZES ]	char(200)	Tételsorhoz tartozó megjegyzés (pl. gyáriszám).
[ CIKKSZAM ]	char(30)	Tétel cikkszám. Cikkszám megadása esetén a tétel

Mező neve	Típus	Leírás
		azonosítása cikkszám alapján történik.
[_VTSZSZJ]	char(11)	Vámtarifaszám, szolgáltatásjegyzék vagy TESZOR szám.
[_AFAKULCS]	char(3)	Csak olyan ÁFA kulcsok adhatók meg, mint amilyenek a számlázó programban is választhatóak.
[_MERTEKEGYSEG]	char(9)	Tétel mértékegysége.
[_MENNYISEG]	Szám	Tétel mennyisége.
[_EGYSEGARNETTO]	Szám	Tétel nettó egységára.
[TERMEKEK_VEGE]		Vége a bizonylat tételeknek.
[BIZONYLATZARO_SZOVEG]	Szöveg (bármilyen hosszú)	A számlán az aláírások felett kinyomtatandó szöveg. A végét a fájl vége vagy egy sor eleji [ karakter jelzi.
[ARFOLYAM]	Szám	Ha a számla nem HUF-os és az ÁFA tartalma nem nulla, akkor ezzel az árfolyammal lesz kiszámítva az ÁFA HUF értéke.
[SORSZAM_VISSZAJELZES]	Char(4)	Alapértelmezett értéke Nem. Megadása esetén a számlázás végén újra meghívja az adatátvitelre beállított php fájlt. Az 'azon' paraméterben az eredeti hivatkozási számot, míg a 'bizonylat_sorszama' paraméterben a kliens oldalon kiállított bizonylat sorszámát adja vissza.
[FILE_VEGE]		A feldolgozás eddig a sorig tart. Ha ez a sor hiányzik, akkor a feldolgozás hibaüzenettel megszakad.

A numerikus adatoknál a tizedes elválasztás lehet pont és vessző is, de az ezres csoportosításnál ezek a karakterek nem használhatóak. Például ez teljesen rossz: 321,321.23. Helyesen: 321321.23 vagy 321321,23.

A \*MOST-al jelölt adatokat csak a ProgramTECH MOST kezeli. Weboldal itt:

<https://most.programtech.hu>

A [TERMEK] blokkot annyiszor kell ismételni, ahány tétel van a számlán.

### Megvalósítás

Mind a fájl alapú, mind a webes illesztés esetében szükséges programozói munka. A fájl illesztés esetében jellemzően a forrás rendszer programozója tud csak Önnek segíteni. Ennek az oka az, hogy általában ezen rendszerek adatbázisai nem elérhetőek külső programok számára.

Webes alkalmazás esetén az adatbázis többnyire elérhető, azokhoz más programozók is tudnak lekérdezéseket készíteni. Természetesen célszerű és praktikus is a forrás rendszer programozóját megkérni a feladatra.

A megvalósítás során gyakoriak a különböző hibák, illetve azok kezelése. Az illesztést végző programozó számára segítség képen készítettünk egy érvényes kimenetet produkáló php oldalt, amin kipróbálhatja a nála futó számlázó program működését, továbbá a kimenetet böngészőben megjelenítve összehasonlíthatja a saját oldalával. A tesztoldal címe és helyes hívása böngészőben:

[https://www.programtech.hu/szamlazo\\_pogram\\_exportdemo.php?hiv\\_szam=123](https://www.programtech.hu/szamlazo_pogram_exportdemo.php?hiv_szam=123)

Megjegyzés: a számlázó programban nem kell beírni a ?hiv\_szam=123 részt. Azt a program majd hozzá teszi a megadott adatok alapján.

Kelt: Szombathely, 2011. december 9.