

Berögztített adatok ellenőrzése

Az ellenőrzés első lépése a szkennelés. Százszor egyszerűbb kikeresni egy adott adatlapot ellenőrzéskor a szkennelt anyagból, mint fizikailag, mivel először a borítékot kell megtalálni az akár 50-100 közül és még azon belül a kérdéses adatlapot. (Amúgy is szkennelni kell az adatlapokat.)

Lépj be az adatbázisba (Nyisd meg böngészőben a könyvjelzők közül a „phpmyadmin” nevűt (<http://172.27.10.229/phpmyadmin/>). Írd be a nagyon bonyolult Felhasználónév – Jelszó párost). A baloldali táblalistából kattints a customer-re.

Kattints az SQL-fülre.

Futtasd le az alábbi lekérdezéseket:

The screenshot shows the phpMyAdmin web interface. On the left, the database list includes 'tesco_form_db (33)'. The main panel shows the 'customer' table selected. The 'SQL' tab is active, and the following query is entered in the text area:

```
SELECT * FROM `customer`  
WHERE (`has_error` = '0' OR `has_error` = '1') AND `form_type` != '2'  
ORDER BY `address_line_1` ASC
```

Below the query area, there is a checkbox for 'Mutasd a parancsot itt újra' (Show the command here again) which is checked. The 'Végrehajt' (Execute) button is visible at the bottom right. On the right side of the SQL editor, a list of fields is shown under the heading 'Mezők száma' (Number of fields):

- id
- name_1
- name_2
- primary_card_account_number
- card_account_number

The 'id' field is selected, and a '<<' button is visible below the list.

➤ Hiányos címek kiszűrése (1.)

Ennek segítségével a lista elejére kerülnek azok a regisztrációk, ahol nincsen település. Ez pl. akkor fordulhat elő, egy olyan irányítószám szerepel, ami nem létezik Magyarországon → keresd ki a szkennelt anyagból, mert lehet, hogy csak el lett ütve. Ha ott is ez az irányítószám szerepel, és nincs helység beírva, akkor jól van így, nem lehet vele mit kezdeni.

Ha olyan irányítószám van település nélkül, amihez több település is tartozik, akkor nézd meg GoogleMaps-szel (<https://www.google.hu/maps/>), hogy az adott utcanév a lehetséges települések közül melyiken van. Ha csak az egyiken → írd be adatbázisba azt a települést; ha 1-nél több településen van ilyen utcanév, akkor marad úgy az adott regisztráció, nem lesz hozzá település.

nily_member_6_dob	address_line_1	address_line_2	address_line_3	province_code	postal_code	mobile_phone_number	email_address
		Kaszap utca	19	9			
	Abadszalók	Dózsa György út	9	9	5241	063-04433332	andrejacquelinehongar
	Akasztó	Tavaszmező utca	3	9	6221		
	Alcsútdoboz	Szabadság utca	193	9	8087		
	Alcsútdoboz	Béke út	43	9	8087		

```
SELECT * FROM `customer`  
WHERE (`has_error` = '0' OR `has_error` = '1') AND `form_type` != '2'  
ORDER BY `address_line_1` ASC
```

➤ Hiányos címek kiszűrése (2.)

Ennek segítségével a lista elejére kerülnek azok a regisztrációk, ahol nincsen irányítószám. Ez pl. akkor fordulhat elő, ha értelmezhetetlen, olvashatatlan a település, irányítószámot pedig nem adott meg a vásárló vagy az is olvashatatlan. Vagy lehet, hogy budapesti címhez nem írtak irányítószámot és az adott utcanév több kerületben is létezik. Ezekkel nem tudunk mit kezdeni.

Mint az alábbi példában látható, a Kürt nevű településhez nem tartozik irányítószám, és Szlovákiában található ez a település. Ezt ilyenkor a szkennelt anyagból ki kell keresni, és az alapján beírni az irányítószámot és módosítani az országkódot.

family_member_6_dob	address_line_1	address_line_2	address_line_3	province_code	postal_code	mobile_phone_number	email
		Kaszap utca	19	9			
	Kürt	Hársfa utca	788	9			
	Budapest	Táncsics Mihály utca	27	9	1014	063-09500713	
	Budapest	Gyula utca	1	9	1016		
	Budapest	Pusztaszeri út	27/a	9	1025		
	Budapest	Törökvész utca	64/a	9	1025	062-05768912	

```
SELECT * FROM `customer`  
WHERE (`has_error` = '0' OR `has_error` = '1') AND `form_type` != '2'  
ORDER BY `postal_code` ASC
```

➤ **Hibás / gyanús házszám, utca**

Ennek segítségével kiszűrhető pl. ha a numerikus billentyűzet használata közben véletlenül becsúszik egy nem kívánt karakter, vagy ha a házszám az utcanévben tárolódik el (rögzítéskor a jellegbe került beírásra), és így üres a házszám mező. További lehetőség, ha hiányzott az utcanév az adatlapon, viszont a jelleg automatikusan felhossa az utcát. Így egy „ utca” lesz bent az utcanév mezőjében, aminek természetesen semmi értelme, ezt ki kell törölni. Előfordulhat olyan eset is, hogy a jelleg mezőt a rögzítő kiütötte egy szóközzel, mert nem volt rá szükség (pl. helyrajziszám, postafiók, egyéb ritka, jelleggel nem rendelkező címek esetében), és ilyenkor egy vagy több felesleges szóköz keletkezik az utcanév végén. Ezeket ki kell törölni.

```
SELECT * FROM `customer`  
WHERE ( (`address_line_3` LIKE '%ű%' OR `address_line_3` LIKE '%+%' OR `address_line_3` LIKE '%-' OR `address_line_3` LIKE '%*%' OR `address_line_2` = '  
utca' OR `address_line_2` LIKE '%ú')  
OR ((`address_line_2` LIKE '%0%' OR `address_line_2` LIKE '%1%' OR `address_line_2` LIKE '%2%' OR `address_line_2` LIKE '%3%' OR `address_line_2` LIKE  
'%4%' OR `address_line_2` LIKE '%5%' OR `address_line_2` LIKE '%6%' OR `address_line_2` LIKE '%7%' OR `address_line_2` LIKE '%8%' OR `address_line_2`  
LIKE '%9%') AND `address_line_3` = '')  
) AND (`has_error` = '0' OR `has_error` = '1')
```

➤ **Néha előfordul, hogy ugyanaz a vonalkód többször szerepel**

- Egy adatlapot 2-szer is berögzítenek (pl. a rögzítő kimegy wc-re, iszik, éppen megszólal a telefonja stb. és elfelejti, hogy a legfelső adatlapot már berögzítette).
- A rögzítő kapkod: egy adatlap rögzítésének végén már elveszi a lapot az egyik kezével és a másik kezével a következőhöz tartozó vonalkódot olvassa be a kézi szkennelrel, majd a következő adatlap rögzítésekor is beolvassa ugyanezt a vonalkódot.
- Egy kártyacserés adatlapra ráírják például azt, hogy „LAKCÍMVÁLTOZÁS”. Ilyenkor 1) berögzítjük a kártyacserét, majd 2) újként felvisszük az adatokat, kártyaszámként ilyen esetben az új (a vonalkódon szereplő) számot adjuk meg. → A vonalkódon szereplő szám ezért kétszer fog szerepelni az adatbázisban.
- Egyszerre érkezik be hozzánk és kerül feldolgozásra egy normál regisztráció és egy ahhoz tartozó kártyacserés regisztráció, mert pl. a vásárló 2-3 nap alatt elhagyta a kártyáját, és lecserélte. → Ez természetesen nem hiba.

Az alábbi SQL-ekkel könnyen kiszűrhető, ha egy vonalkód többször szerepel.

Többször szereplő primary_card_account_number (Mikor lesz egynél többször egy adott kártyaszám a találati listában? Például: tévedésből duplán rögzített adatlap; a rögzítő kapkodott; eredeti és cserés regisztráció is egyszerre szerepel):

```
SELECT `primary_card_account_number`, `stored_time` AS `nap`, COUNT(*) AS `össz` FROM `customer`  
WHERE `primary_card_account_number` != ""  
AND (`has_error` = '0' OR `has_error` = '1')  
GROUP BY `primary_card_account_number`  
ORDER BY `össz` DESC, `nap` DESC
```

Többször szereplő card_account_number (Mikor lesz egynél többször egy adott kártyaszám a találati listában? Például: tévedésből duplán rögzített adatlap; lakcímváltozás miatt „újként” is berögzítünk egy kártyacserés regisztrációs lapot):

```
SELECT `card_account_number`, `stored_time` AS `nap`, COUNT(*) AS `össz` FROM `customer`  
WHERE `primary_card_account_number` != ""  
AND (`has_error` = '0' OR `has_error` = '1')  
GROUP BY `card_account_number`  
ORDER BY `össz` DESC, `nap` DESC
```

A duplán rögzített adatlapok mindkét előbbi lekérdezésben megjelennek, mivel ott megegyezik a *primary_card_account_number* és a *card_account_number* mezők értéke.

Találtunk egy duplán rögzített adatlapot:

Adatbázis
_form_db (33)

tesco_form_db (33)

- customer
- customer_arch20100920_1
- customer_arch20100920_2
- customer_arch20100920_3
- customer_arch20100920_4
- customer_arch20100920_5
- customer_arch20100920_6
- customer_arch20100920_7
- customer_arch20100920_8
- customer_arch20101015_1
- customer_arch20101130_1
- customer_arch20110202
- customer_arch20110331
- customer_arch20110901
- customer_arch20120101
- domain_lista
- jelleg
- kulfold_20110207
- nevek_kereszt
- nevek_vezetek
- skeleton_20110208
- stat
- stat_info
- telepulesek
- telepules_budapest
- telepules_debrecen
- telepules_gyor
- telepules_miskolc
- telepules_pecs
- telepules_szeged
- temp
- users
- utcanev

Sorok megjelenítése 0 - 29 (7 502 Összesen, A lekérés lefutott 0.2624 másodperc alatt)

SQL-kérés:

```
SELECT `card_account_number` , `stored_time` AS `nap` , COUNT( * ) AS `ossz`  
FROM `customer`  
WHERE `primary_card_account_number` != ''  
AND (  
  `has_error` = '0'  
  OR `has_error` = '1'  
)  
GROUP BY `card_account_number`  
ORDER BY `ossz` DESC , `nap` DESC  
LIMIT 0 , 30
```

☐ Profiling [Szerkeszt] [SQL magyarázat]

Mutat: 30 sor kezdve ezzel: 30

vízszintes módon, a fejléct 100 soronként megismételve

> >>

Oldalszám: 1

primary_card account_number nap ossz

634009540019013103 2012-05-08 2

634009540010820704 2012-05-08 1

634009540019156639 2012-05-08 1

634009540019156670 2012-05-08 1

634009540019558065 2012-05-08 1

634009540018957888 2012-05-08 1

634009540018958084 2012-05-08 1

634009540019773110 2012-05-08 1

634009540019820663 2012-05-08 1

634009540020128023 2012-05-08 1

Megnézzük a kérdéses vonalkódhoz tartozó adatokat.

Láthatjuk, hogy ugyanaz az adatlap lett 2-szer berögzítve:

SQL-keres:

```
SELECT *  
FROM `customer`  
WHERE `primary_card_account_number` LIKE '634009540019013103'  
LIMIT 0 , 30
```

☐ Profiling [Szerkeszt] [SQL magyarázat] [PHP kód készítése] [Frissítés]

Mutat: 30 sor kezdve ezzel: 0

vízszintes módon, a fejléct 100 soronként megismételve

Kulcs szerinti rendezés: Nincs

	id	name_1	name_2	primary_card_account_number	card_account_number	family_member_1_gender_code	family_member_1_dob	fa
<input type="checkbox"/>	1489429	Vágvölgyi	Sándor	634009540019013103	634009540019013103	M	1933-02-18	
<input type="checkbox"/>	1489430	Vágvölgyi	Sándor	634009540019013103	634009540019013103	M	1933-02-18	

↑ Összeset kijelöli / Összeset törli A kijelöltekkel végzendő művelet: ☐ ☒ ☐

Mutat: 30 sor kezdve ezzel: 0













vízszintes módon, a fejléct 100 soronként megismételve

Ki kell törölni az egyiket.

➤ Program által hibásnak jelölt (has_error = 1)

```
SELECT * FROM `customer`  
WHERE `has_error` = '1'
```

Hiányzó vonalkód miatt lehet 1-es a has_error, de ezt NEM SZABAD módosítani, mivel XML-írásig ennek a hibakódnak 1-esnek kell maradnia, különben nem kerülne bele a MissingCC-s file-ba az adott regisztráció. Az XML kiírása után automatikusan 2 lesz a has_error értéke.

			id	name_1	name_2	primary_card_account_number	card_account_number	family_member_1_gender_code	family_member_1_dob	fa
<input type="checkbox"/>			2744410	Kótai	Nikolett Etelka			F	1988-12-31	
<input type="checkbox"/>			2744490	Vámosi	Zoltánné			F	1956-02-01	
<input type="checkbox"/>			2744491	Duczár	Andrea			F	1979-11-25	
<input type="checkbox"/>			2744950	Porkoláb-Minarik	Annamária			F	1975-09-11	
<input type="checkbox"/>			2745037	Takács	Adrienn Anita			F		
<input type="checkbox"/>			2745357	Nagy	Imre			M	1949-08-05	

Ha a rögzítő véletlenül hagyta ki egy regisztráció vonalkódját, akkor pótolni kell mindkét (primary_card_account_number és card_account_number) mezőben, valamint a has_error mező értékét 1-ről 0-ra kell állítani, így az XML kiírásakor a megfelelő fájlba fognak kerülni a regisztráció adatai.