

IBMiSqlScripts

Zpracování SQL skriptů

Návod k použití

Obsah

Obsah	2
Úvod	4
Použití aplikace	5
<i>Adresáře</i>	5
<i>Programové soubory</i>	5
<i>Umístění a spouštění aplikace</i>	5
<i>Instalace nové verze</i>	5
Obrazovky pro správce aplikace	6
<i>Export</i>	6
<i>Import</i>	8
<i>Parametry</i>	9
Jazyk aplikace	9
Adresa serveru	9
Jméno uživatele	10
Seznam knihoven s databázovými tabulkami	10
Adresář IFS k centrálnímu uložení skriptů	10
Velikost okna s výsledky skriptu (dotazu)	10
Značka pro prázdné hodnoty sloupců	11
Počet mezer oddělujících sloupce ve výsledku dotazu	11
Výška písma v počtu tiskových bodů	11
Maska pro výstup čísel	11
<i>Editace</i>	12
Vytvořit nový skript	13
Obnovit zobrazení	13
Uložit do serveru	13
Načíst ze serveru	13
Zrušit vybraný	13
Upravit vybraný	14
Obrazovky aplikace pro uživatele	15
<i>Spuštění</i>	15
<i>Parametry</i>	15
Spouštění skriptů	16
<i>Dotaz bez proměnných parametrů</i>	16
<i>Dotaz s proměnnými parametry</i>	19
<i>Nedotazový skript</i>	21
Pravidla pro tvorbu skriptů	24
<i>Definice proměnných hodnot parametrů (parameter markers)</i>	24
Údaje v definičním řádku	25
Zadání hodnot parametrů při spouštění skriptu	26
Příklad zadání parametrů	27
<i>Definice titulních hlaviček</i>	29
<i>Definice sloupcových hlaviček</i>	30
Vlastní hlavičky	30
Standardní hlavičky	30
<i>Definice vertikálního a horizontálního členění</i>	31
<i>Úprava čísel pro výstup</i>	33
Symboly pro tvorbu masek	33
Příklady masek	33
Zadání masek	35
Formát čísel bez masky	36
<i>Vynechání sloupců z výstupu</i>	37
Příklad na vynechání sloupců a tisk titulních hlaviček	37

<i>Parametry pro tisk</i>	39
<i>Sumarizace výsledků dotazu</i>	40
Definice skupinových úrovní	40
Definice sumarizovaných sloupců	40
Úvodní texty pro sumární řádky	41
Příklad 1	42
Příklad 2	45
Editace skriptů	48
<i>Hledání textu</i>	49
<i>Shifting selected text</i>	50
Horizontal selection	50
Vertical selection	50
<i>Copy, cut and paste selected text</i>	51
Horizontal selection	51
Vertical selection	51
<i>Column lists – Seznamy sloupců</i>	52
Odstranění vybraných sloupců	52
Sestavení seznamu sloupců	53
Sestavení seznamu sloupců do řádku	54
Výběr všech jmen v levém i pravém rámečku	54
Kontextová nabídka na pravém rámečku	54
Práce s členy fyzických a logických souborů	55
<i>Vytvoření fyzického a logického souboru se členy</i>	55
<i>Skript k vytvoření alias objektů a zápisu dat do členů</i>	56
<i>Kontrolní výpis dat ze členů logického souboru</i>	57

Úvod

Motivem k vytvoření této aplikace bylo zjištění, že populární program Query/400 (s pozdějším názvem Query for i) není schopen zobrazovat a tisknout všechny znaky kódované v soustavě UCS-2 (CCSID 13488), UTF-16 (CCSID 1200) nebo UTF-8 (CCSID 1208).

Aplikace dovoluje vytvářet, ukládat a spouštět SQL skripty pro databázi IBM DB2 for i. Skript je text příkazů jazyka SQL uložený v textovém souboru s koncovkou `.sql` nebo `.SQL`. Do skriptu lze zapsat jeden nebo více SQL příkazů oddělených středníkem. Příkazy mohou být libovolného druhu a mohou obsahovat parametry označené otazníky. Nejčastěji ovšem půjde o dotazy, to znamená příkazy SELECT. Z uvedeného plyne, že tvůrce skriptů musí znát jazyk SQL alespoň na úrovni příkazu SELECT.

Do skriptů se kromě SQL příkazů obvykle doplňují ještě další příkazy ve formě speciálních komentářových řádků, které dovolují pružně formátovat výstup výsledků na obrazovce a tisku na papír.

Programy jsou napsány v jazyku Java a vyžadují verzi *Java SE 8* nebo vyšší. Spolupracují s programy soustavy *IBM i Toolbox for Java* (nebo JTOpen). Programy byly vytvořeny a testovány v systémech Mac OS X, macOS a Windows 7, 10, 11 se vzdáleným internetovým připojením k systému IBM i.

Aplikace se neinstaluje, je připravena k okamžitému použití (viz [Použití aplikace](#) níže).

Použití aplikace

Aplikace se neinstaluje, je celá předávána jako adresář, v němž jsou začleněny následující adresáře a soubory.

Poznámka: Zde neuvedené objekty, které jsou v aplikačním adresáři navíc, slouží k vedení projektu a mohou být beze škody odstraněny.

Adresáře

- *helpfiles* - obsahuje návod k použití v českém a anglickém jazyce,
- *icons* - obsahuje ikony pro tlačítka v editoru,
- *logfiles* - obsahuje textové soubory *err.txt* a *out.txt*, do nichž se zapisuje přesměrovaný výstup ze souborů *System.err* a *System.out* (tj. z konzoly).
- *paramfiles* - obsahuje soubor *Parameters.txt* s parametry aplikace,
- *printfiles* - obsahuje textové soubory, které jsou výsledkem běhu skriptů,
- *scriptfiles* - obsahuje soubory skriptů, jejichž jména končí příponou *.sql*,
- *workfiles* - obsahuje textový soubor *Print.txt*, výsledek posledního běhu skriptu.

Poznámka: Soubory *err.txt* a *out.txt* slouží ke zjištění původu případné chyby v programu.

Programové soubory

- Soubor *jt400-21.0.4.jar* - obsahuje podmnožinu tříd ze soustavy IBM i Toolbox for Java.
- Soubor *Q_Menu.jar* - obsahuje třídy Java programů aplikace a spouští aplikaci pro správce aplikace a pro tvůrce skriptů.
- Soubor *Q_MenuUser.jar* - obsahuje třídy Java programů aplikace a spouští aplikaci pro uživatele.

Umístění a spouštění aplikace

Aplikační adresář umístíme na vhodné místo a případně přejmenovat. V něm můžeme vytvořit zástupce (shortcut, alias) souborů *Q_Menu.jar* (programy pro správce) a *Q_MenuUser.jar* (programy pro uživatele). Zástupce pak můžeme umístit třeba na plochu nebo jinam.

Programy spustíme poklepáním na zástupce (nebo na originály) souboru *.jar*; ukáže se nabídka aplikace.

Stejná kopie aplikace funguje v systémech macOS a Windows.

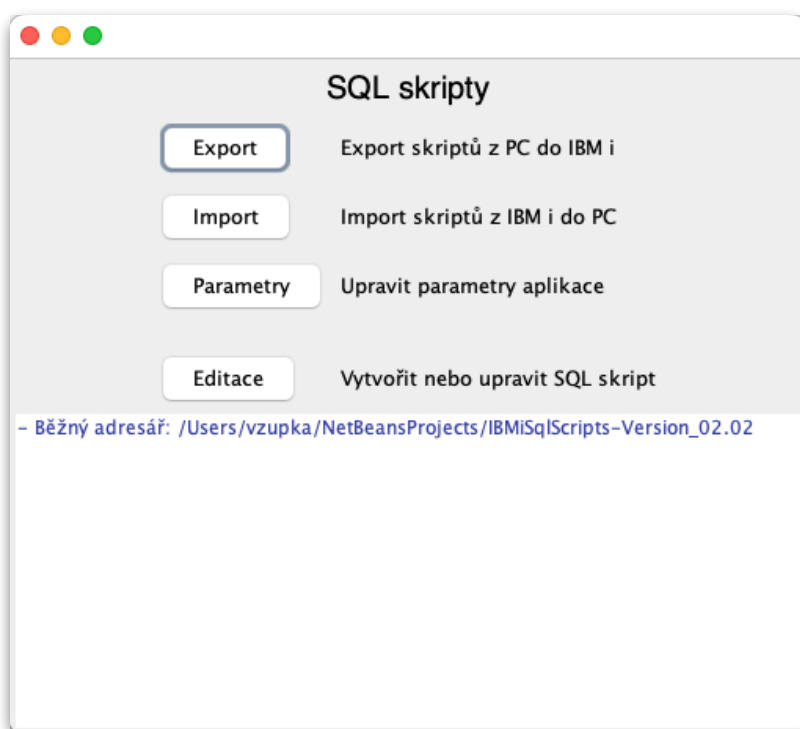
Instalace nové verze

Nahradíme dva *.jar* soubory *Q_Menu.jar* a *Q_MenuUser.jar* jejich novými verzemi. Nahradíme také obsah adresáře "helpfiles" se soubory *IBMiSqlScriptsUserDocEng.pdf* a *IBMiSqlScriptsUserDocCz.pdf* obsahujícími uživatelskou dokumentaci.

Upozornění: Adresář *scriptfiles* obsahuje texty skriptů, které vytvořil správce aplikace. Podle potřeby je lze zapsat do jiného adresáře nebo do adresáře *skriptfiles* přechíst skripty uložené v jiném adresáři. Trvaleji lze uschovat skripty do IFS adresáře, který správce vytvoří k tomuto účelu. Může jich vytvořit několik a podle potřeby je použít k exportu z adresáře *scriptfiles* nebo k importu do adresáře *skriptfiles*, viz [dále](#).

Obrazovky pro správce aplikace

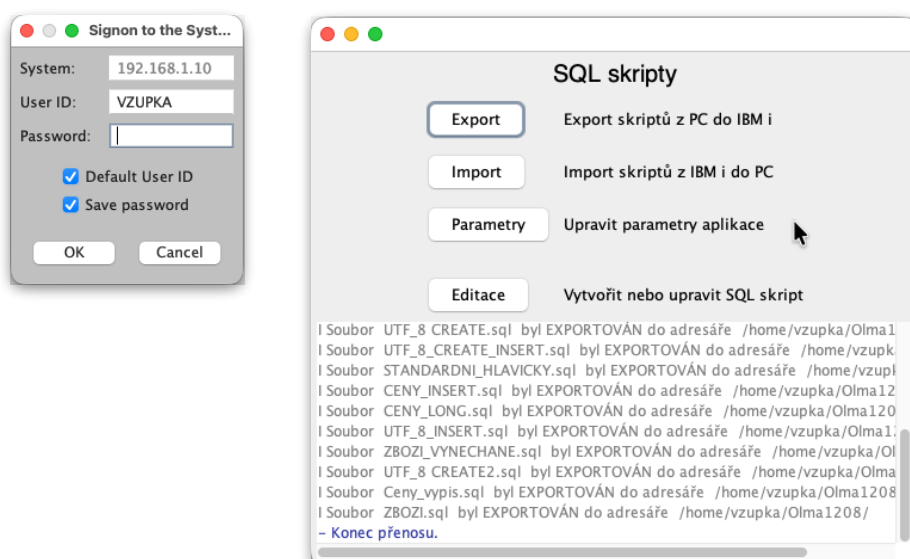
Při spuštění aplikace se jako první zobrazí nabídka funkcí pomocí tlačítek.



Export

Tato funkce vyvolá dialog *Signon to the System* pro zadání přihlašovacích údajů k přístupu do systému IBM i.

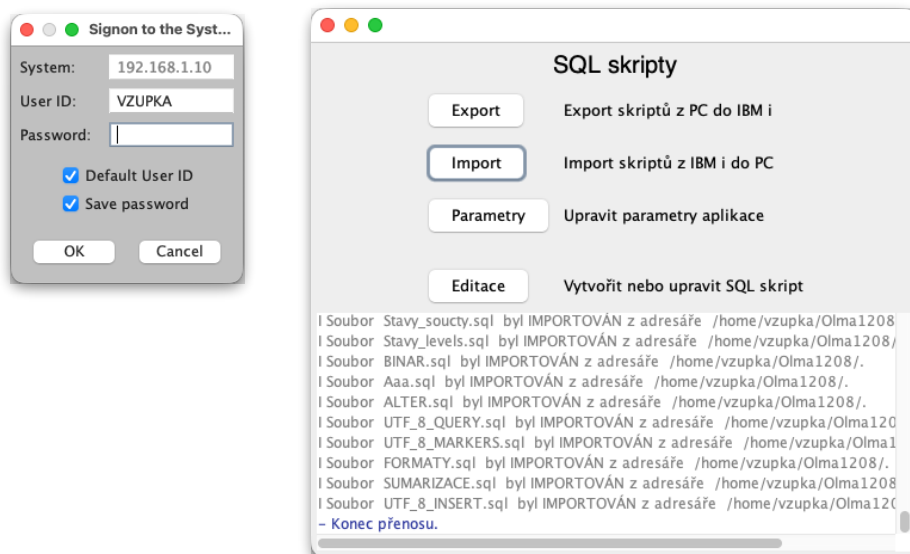
Po úspěšném přihlášení spustí přenos *všech skriptů* z lokálního adresáře *scriptfiles* do adresáře IFS zadaného v parametrech aplikace. O přenosu vypíše protokol.



Import

Tato funkce vyvolá dialog *Signon to the System* pro zadání přihlašovacích údajů k přístupu do systému IBM i.

Po úspěšném přihlášení spustí přenos *všech skriptů* z adresáře IFS zadaného v parametrech aplikace do lokálního adresáře *scriptfiles*. O přenosu vypíše protokol.



Parametry

Zadání parametrů aplikace

English ☐ Application language. Restart the application after change.
Česky ☒ Jazyk aplikace. Po změně spusťte aplikaci znovu.

192.168.1.10 Adresa serveru

VZUPKA Jméno uživatele

CORPDATA, VZTOOL Seznam knihoven s databázovými tabulkami

/home/vzupka/Olma1208 Adresář IFS k centrálnímu uložení skriptů

☒ Automatická velikost okna s výsledky dotazu

450 Šířka okna s výsledky dotazu

450 Výška okna s výsledky dotazu

- Značka pro prázdné hodnoty sloupců, není-li ve skriptu zadána jiná

1 Počet mezer oddělujících sloupce ve výsledku dotazu

40 Výška písma v počtu tiskových bodů

#0.00 Masky pro tisk čísel, např. #.00 pro potlačení nul zleva a zachování dvou desetinných míst

Uložte data Návrat

- Běžný adresář: /Users/vzupka/NetBeansProjects/IBMiSqlScripts-Version_02.02

Uživatel zadá jméno svého profilu pro přístup k systému IBM i. Toto jméno pak bude předvoleno v přihlašovacím dialogu *Signon to the System*:

Signon to the Syst...

System: 192.168.1.10

User ID: VZUPKA

Password:

☒ Default User ID

☒ Save password

OK Cancel

Jazyk aplikace

Aplikace může být provozována v anglické (en_US) nebo české (cs_CZ) lokalizaci. Lokalizace se týká nadpisů, zpráv, označení tlačítek, úpravy čísel pro výstup výsledků a úpravy data a času v hlavičce výstupu. Uživatel si může zvolit, ve které lokalizaci bude aplikaci provozovat. Po změně jazyka se volba plně uplatní (včetně úvodní obrazovky), když uživatel ukončí aplikaci a spustí ji znovu.

Adresa serveru

Je nutné zadat jedinou IP adresu v podobě tečkové nebo doménové.

Jméno uživatele

Správce zadá jméno profilu, který má oprávnění zapisovat soubory do IFS adresáře. Bude-li provádět manipulační SQL příkazy (jiné než SELECT), musí mít oprávnění pro zápis a přepis dat v databázových tabulkách (souborech), popřípadě k manipulaci s celými tabulkami (vytváření, rušení).

Seznam knihoven s databázovými tabulkami

Aplikace používá ke zpracování SQL příkazů jmennou konvenci “system”. Správce zapíše jedno nebo několik jmen knihoven oddělených čárkou nebo mezerou, anebo žádné jméno.

V době připojení databáze k programu se vytvoří seznam knihoven, který nahradí seznam z úlohy databázového serveru (server job). Při zpracování SQL příkazu se prohledává seznam knihoven v zapsaném pořadí. Je-li prvním jménem *LIBL, připojí se zadaný seznam k seznamu z úlohy serveru.

Upozornění: Přednost má ovšem jméno knihovny (schematu) zadané přímo v SQL příkazu jako kvalifikátor před jménem tabulky (nebo jiného objektu). Oddělovač ve kvalifikaci může být *tečka* (.) nebo *lomítko* (/). Jakmile je takový příkaz použit, další příkazy běžící v pořadí hledají objekty pouze v této knihovně (schématu). Nenaudou-li je tam, systém to ohlásí chybovou zprávou. Pro nápravu lze použít dvojí opatření:

- Na konci skriptu zapíšeme příkaz SET SCHEMA DEFAULT, který aktivuje původní seznam knihoven.
- Vytvoříme skript s jediným příkazem SET SCHEMA DEFAULT a provedeme jej předtím než spustíme příkaz, který ohlásil nenalezení požadovaného objektu.

Adresář IFS k centrálnímu uložení skriptů.

Adresář IFS slouží jako centrální úložiště k ukládání (export) a načtení (import) skriptů do/ze serveru IBM i. Není-li zadána cesta k adresáři, nelze tyto funkce používat. Cesta k adresáři musí začínat lomítkem / a může, ale nemusí lomítkem končit. Aplikace si zajistí sama správnost jejího zakončení.

Adresář musí být předem vytvořen pomocí CL příkazu CRTDIR (nebo MKDIR, MD), např.

```
CRTDIR DIR(' /home/vzupka/Olma1208')
```

V adresáři budou uloženy texty skriptů v kódování UTF-8. Toto kódování lze určit pro adresář i všechny jeho soubory příkazem

```
CHGATR OBJ(' /home/vzupka/Olma1208') ATR(*CCSID) VALUE(1208)
```

Usnadní se tak pro prohlížení skriptů v emulaci 5250 pomocí CL příkazu

```
WRKLNK ' /home/VZUPKA/Olma1208'
```

Velikost okna s výsledky skriptu (dotazu)

Je-li zaškrtnut čtvereček “Automatická velikost okna s výsledky dotazu”, okno se přizpůsobí rozměrům zobrazených výsledků. V opačném případě bude okno zobrazeno v rozměrech uvedených v polích “Šířka okna s výsledky dotazu” a “Výška okna s výsledky dotazu” s případnými posuvníky.

Poznámka: Není-li vstupní hodnota celé číslo, změní se na 0.

Značka pro prázdné hodnoty sloupců

Text zapsaný v tomto poli (případně prázdný znak) se bude zobrazovat (a tisknout) všude, kde je výsledná hodnota sloupce prázdná (NULL).

Poznámka: Text pro prázdnou hodnotu sloupce lze také zadávat individuálně pro jednotlivé skripty pomocí komentářového definičního řádku (viz dále).

Počet mezer oddělujících sloupce ve výsledku dotazu

Tento údaj je číslo udávající počet mezer, které se zařadí na konec každého sloupce v řádcích hlaviček i v řádcích dat ve výsledném textu dotazu. Může být i 0. Není-li vstupní hodnota celé číslo, dosadí se 0.

Poznámka: Počet mezer lze také zadat individuálně pro jednotlivé skripty pomocí komentářového definičního řádku (viz dále).

Výška písma v počtu tiskových bodů

Tento údaj představuje počet bodů, kde jeden bod měří 1/72 palce (inch). Využije se u tisku výsledků na papír. Není-li vstupní hodnota celé číslo, dosadí se 0.

Poznámka: Výšku písma lze také zadat individuálně pro jednotlivé skripty pomocí komentářového definičního řádku (viz dále).

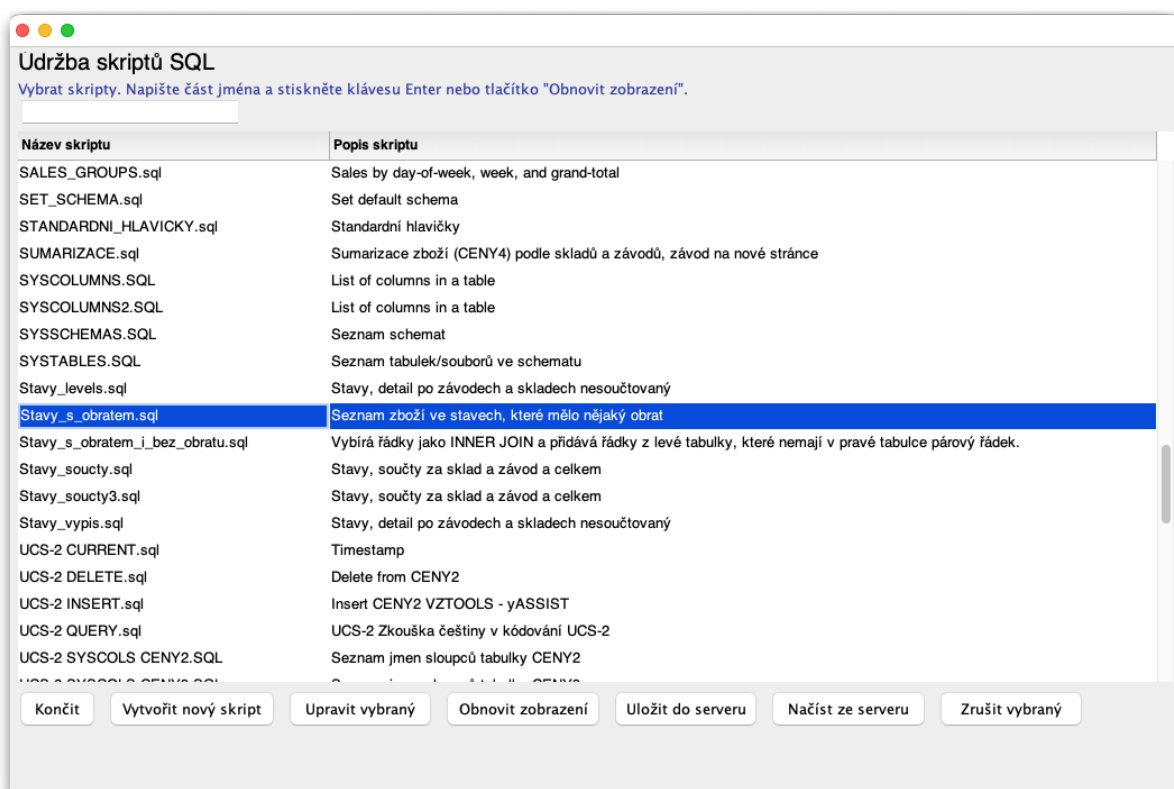
Maska pro výstup čísel

Předvolená maska pro výstup čísel určuje výstupní obraz dekadických čísel ve všech skriptech. Nejpoužívanější bude asi maska #0.00, která potlačuje nuly zleva, ale nechává nulu před desetinným oddělovačem (čárkou) a zachovává dvě desetinná místa včetně koncových nul. Je-li zadána prázdná hodnota, uplatní se buď masky zadané pro jednotlivé sloupce nebo pravidlo pro standardní výstup čísel (viz dále).

Poznámka: Masky lze zadávat i jednotlivě pro zvolené sloupce výsledné tabulky, a to pomocí komentářových definičních řádků v každém skriptu.

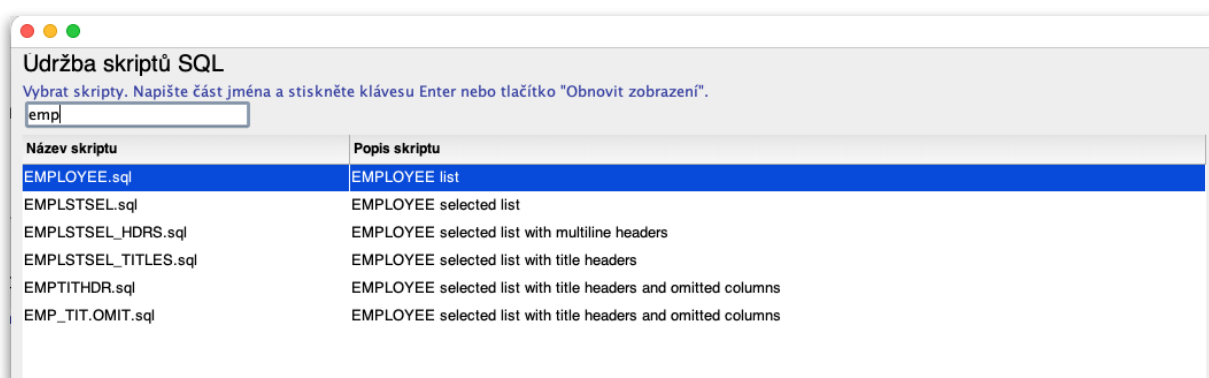
Editace

Zobrazí tabulku se seznamem skriptů, který je zpočátku prázdný. Je-li neprázdný, zobrazit se celý, je-li výběrové textové pole prázdné.



Uživatel seznam omezit tak, že do textového pole napíše *výběrový vzorek* a pak stiskne klávesu *Enter* nebo tlačítko *Obnovit zobrazení*.

Výběrový vzorek je libovolný text, který se hledá ve jménech všech skriptů z adresáře *scriptfiles*. Při výběru se ignoruje velikost písmen.

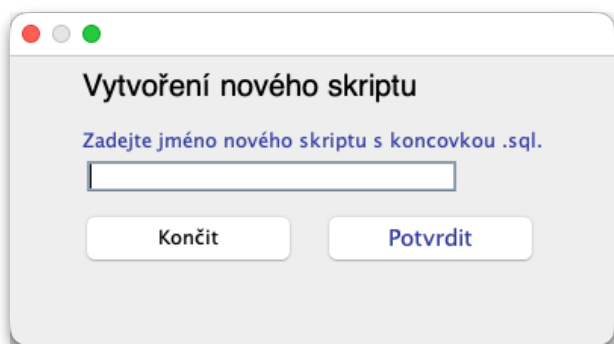


Zde se vybrala jména začínající EMP.

V okně je řada tlačítek: *Končit*, *Vytvořit nový skript*, *Obnovit zobrazení*, *Načíst ze serveru* a *Zrušit vybraný*. Ta jsou popsána dále.

Vytvořit nový skript

Tlačítko vyvolá okno se vstupním polem, do něhož uživatel zapíše zvolené jméno zakončené koncovkou `.sql`. a potvrdí tlačítkem *Potvrdit* nebo klávesou *Enter*.



Jestliže takový skript již existuje, ohlásí se chybová zpráva.

Poznámka: Jméno skriptu nesmí být prázdné a smí obsahovat jen znaky kódu ASCII a musí končit příponou `.sql`.

Obnovit zobrazení

Smaže případnou zprávu a zruší výběr řádku ze seznamu. Je-li v textovém poli zadán text, zobrazí v tabulce jen skripty, v jejichž jméně je text obsažen (bez ohledu na velikost písmen). Je-li text prázdný, zobrazí všechny skripty.

Uložit do serveru

Není-li připojen server, vyvolá okno pro zadání přihlašovacích údajů. Po úspěšném přihlášení spustí přenos *vybraného skriptu* z lokálního adresáře *scriptfiles* do adresáře IFS zadaného v parametrech. O přenosu zobrazí zprávu.

Načíst ze serveru

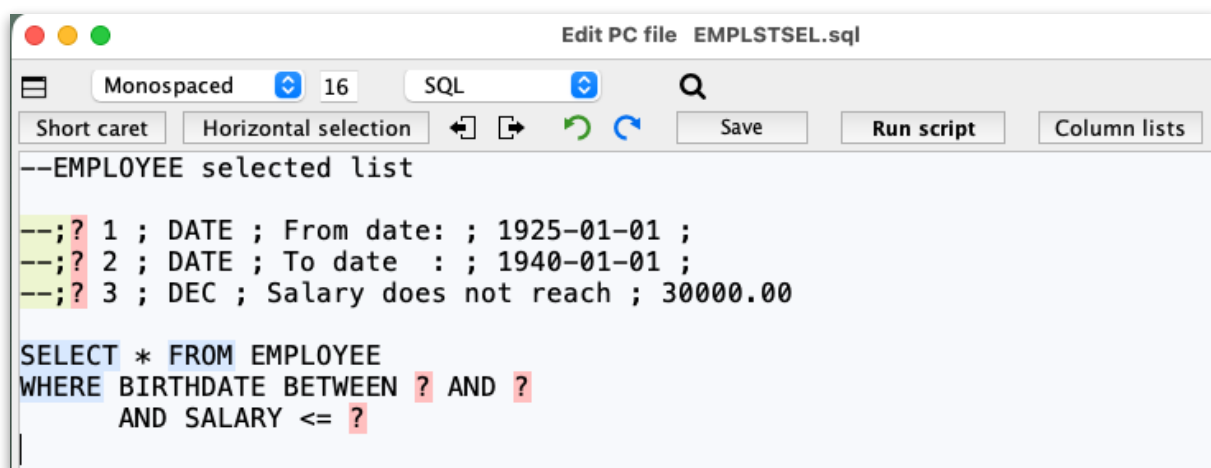
Není-li připojen server, vyvolá okno pro zadání přihlašovacích údajů. Po úspěšném přihlášení spustí přenos *vybraného skriptu* z adresáře IFS zadaného v parametrech do lokálního adresáře *scriptfiles*. O přenosu zobrazí zprávu.

Zrušit vybraný

Skript z vybraného řádku se rovnou zruší v lokálním adresáři *scriptfiles* (bez kontrolního dotazu). Dále se zobrazí okno pro zadání přihlašovacích údajů k přístupu do systému IBM i. Po úspěšném přihlášení se skript zruší i v adresáři IFS. Jestliže přihlášení nedokončíme (zrušíme), program ohlásí chybovou zprávu a skript v adresáři IFS zůstane zachován (v případě, že tam vůbec je). Není-li vybrán žádný řádek, nic se neruší, jen se o tom zobrazí zpráva.

Upravit vybraný

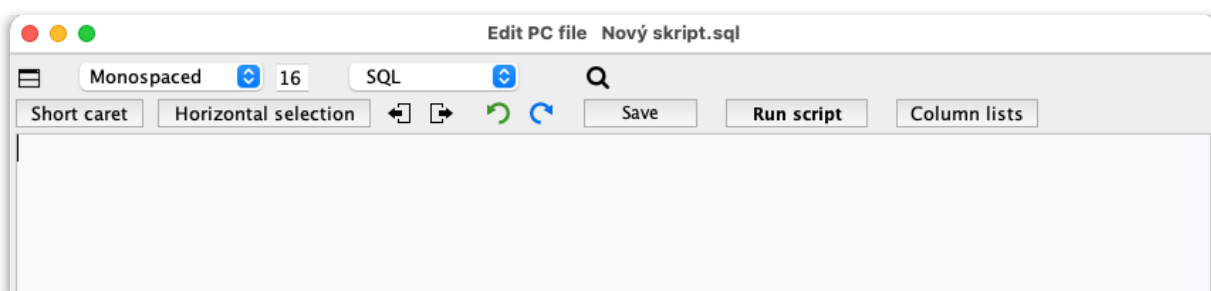
Vybraný skript (např. EMPLSTSEL) se zobrazí v novém okně a uživatel může zapsat nebo editovat text skriptu a pak stisknout tlačítko *Save* nebo klávesovou zkratku *Cmd S*.



Note: *Cmd* key is used in macOS instead of Ctrl.

Skript je uložen do svého souboru v adresáři *scriptfiles*.

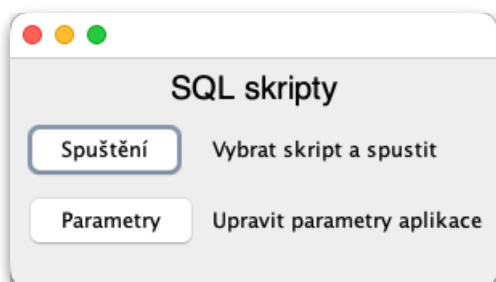
Je-li skript nový, editor obsahuje prázdnou oblast.



Uživatel pak může zapsat celý text skriptu nebo jej zkopírovat z jiného zdroje a vložit (copy/paste) (viz kapitolu [Editace skriptů](#)).

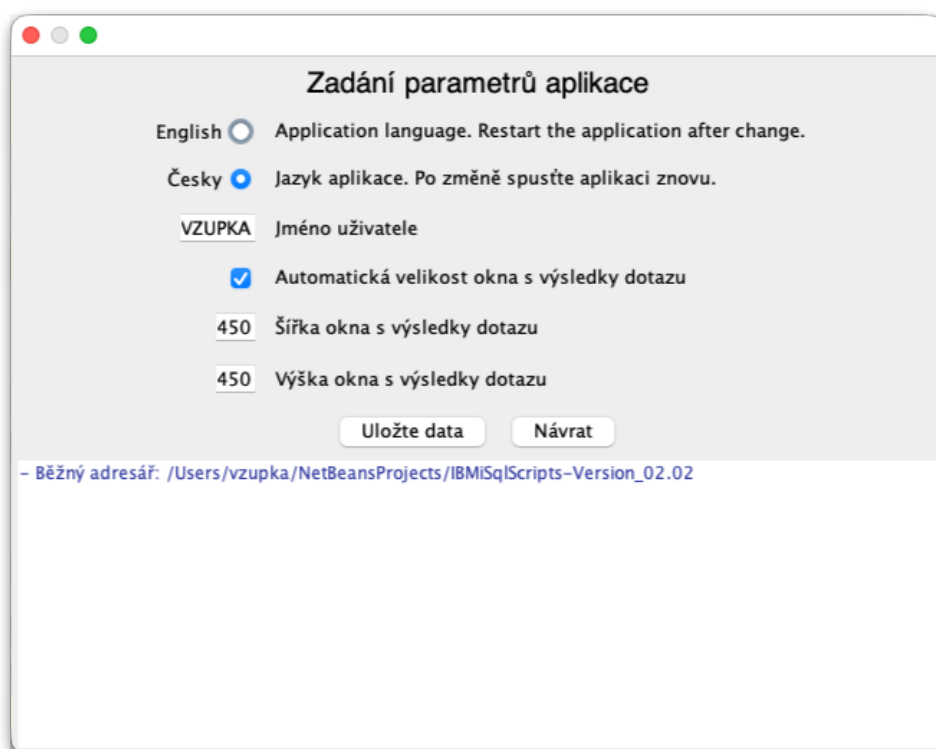
Obrazovky aplikace pro uživatele

Tato hlavní obrazovka je zkrácená proti té, která je určena pro správce aplikace a tvůrce skriptů.



Spuštění

Tato funkce je popsána dále v kapitole *Spouštění skriptů*.



Parametry

Tato funkce je zjednodušenou verzí funkce určené pro správce aplikace a tvůrce skriptů (viz výše). Nabízí jen ty parametry, které využije běžný uživatel. Mají stejný význam jak bylo popsáno výše.

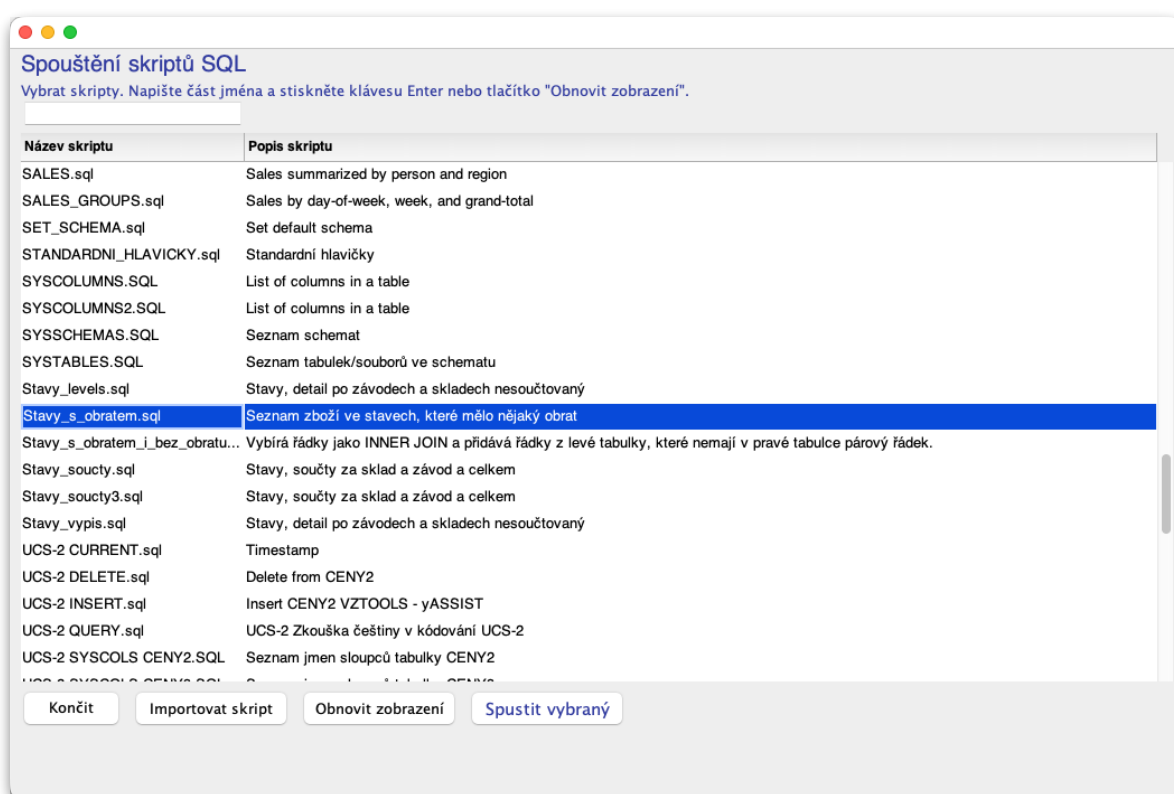
Spouštění skriptů

V této kapitole je spouštění skriptů ilustrováno na příkladech. Předpokládáme, že skripty zde uvedené byly dříve vytvořeny administrátorem. Uživatel text skriptu nevidí, vidí jen jeho popis a pak výsledek.

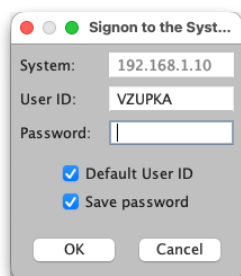
Z hlavní nabídky aplikace se po stisku tlačítka *Spuštění* zobrazí okno s nadpisem “Spouštění skriptů SQL” obsahující seznam skriptů. Zpočátku je seznam skriptů prázdný. Aby se do něj dostaly záznamy, musí správce aplikace vytvořit skripty pomocí [funkce Editace](#) (viz níže).

Dotaz bez proměnných parametrů

V seznamu vybereme např. řádek s názvem skriptu *Stavy_s_obratem* a tlačítkem *Spustit vybraný* spustíme dotaz.

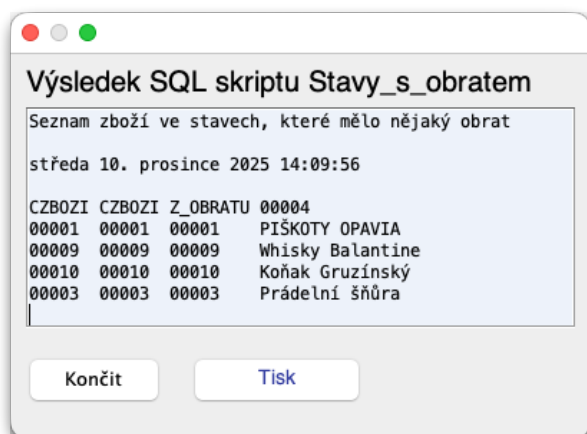


Jestliže nebyl připojen server, objeví se přihlašovací okno, kde zadáme heslo.



Po zadání hesla výpočet pokračuje.

Výsledek dotazu se zobrazí v okně.



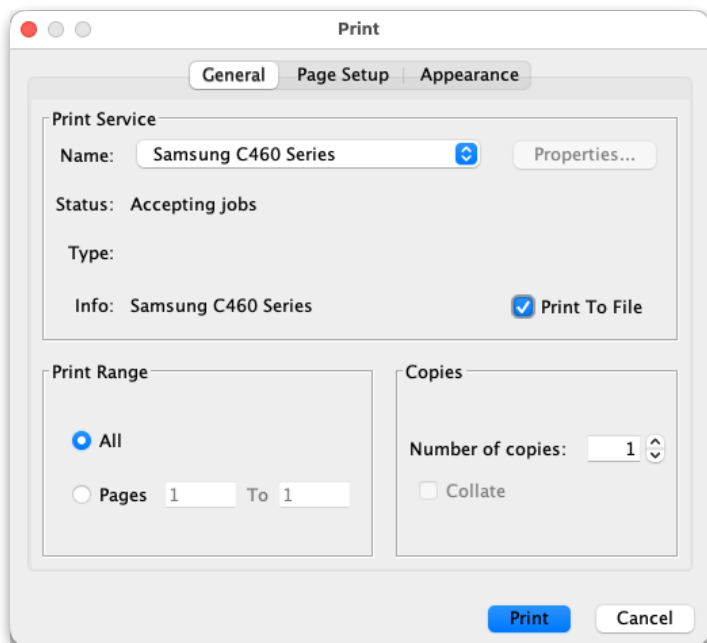
Výsledek se zároveň zapíše do souboru *Print.txt* v adresáři *workfiles* a navíc do souboru *Stavy_s_obratem.txt* v adresáři *printfiles*. V hlavičce nad sloupci dat jsou standardní nadpisy shodné se jmény sloupců v příkazu SQL.

Skript, který ovšem uživatel nevidí, je následující.

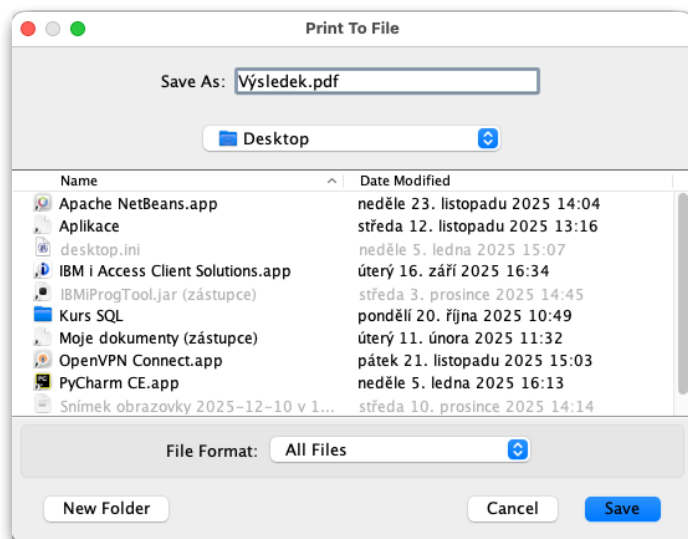
```
--Seznam zboží ve stavech, které mělo nějaký obrat
SELECT DISTINCT C.CZBOZI, S.CZBOZI, O.CZBOZI Z_OBRATU, VARCHAR(C.NAZZBO, 30)
FROM CENY4 C
  INNER JOIN STAVY S ON C. CZBOZI = S. CZBOZI
  INNER JOIN OBRATY O ON S. CZBOZI = O. CZBOZI
```

Nyní se uživatel může spokojit s výsledkem a ukončit dotaz, nebo může výsledky vytisknout na tiskárně použitím tlačítka *Tisk*.

K provedení tisku se zobrazí dialogové okno podobné tomuto.

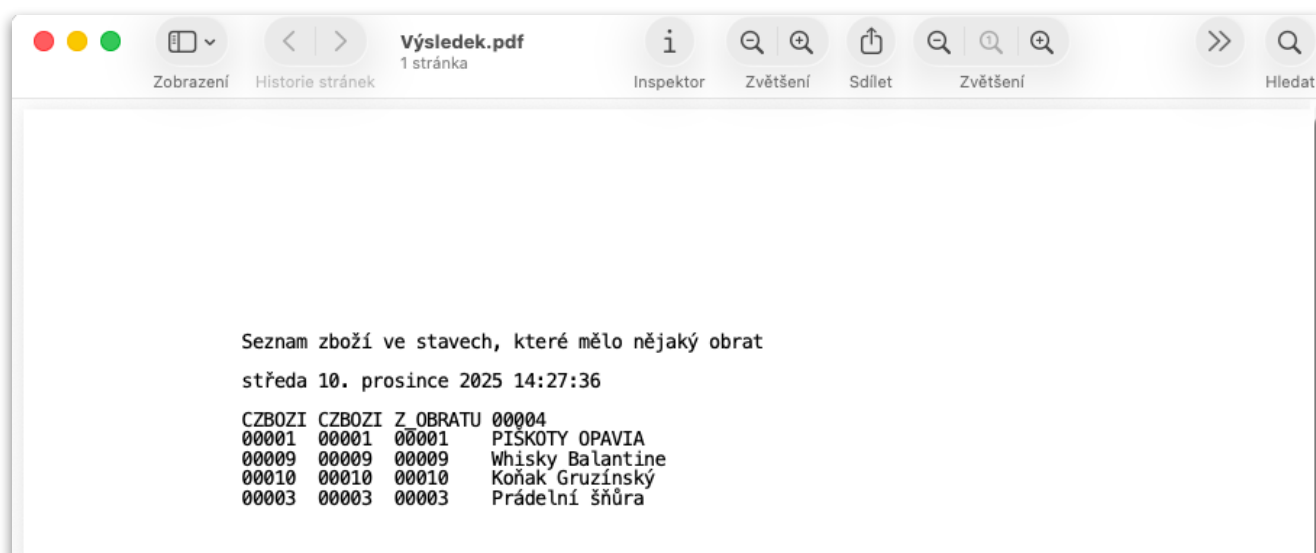


Zaškrtneme-li políčko *Print to file*, pak stiskem tlačítka *Print* místo tisku na papír vyvoláme nabídku adresářů.



Z nabídky vybereme vhodný adresář, například *Desktop*, kam můžeme uložit výsledek fláčkem *Save*, třeba jako soubor *Výsledek.pdf*.

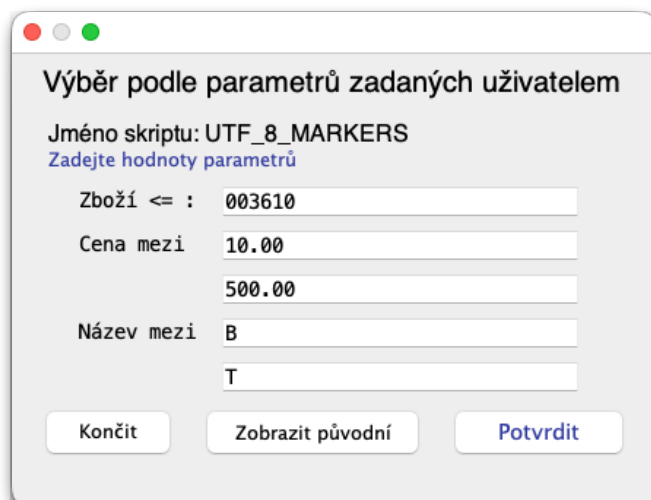
Ukázka části stránky PDF souboru (macOS):




Dotaz s proměnnými parametry


Příkaz SQL může obsahovat parametry označené otazníkem. V seznamu vybereme například řádek s názvem skriptu UTF_8_MARKERS.sql.


Stiskem tlačítka *Spustit* vybraný se zobrazí okno s výzvou k zadání hodnot pro tyto parametry.



Předvolené údaje můžeme změnit nebo ponechat. Po stisku tlačítka *Potvrdit* se zobrazí okno s výsledkem dotazu.

Necháme-li parametry nezměněné, zobrazí se následující výsledek. Můžeme si všimnout obrázku míče  (Unicode: U+26BD U+FE0F, UTF-8: E2 9A BD EF B8 8F).

; 00012 45,00 Pálka pingpongová; 00006 10,55 Ponožky pánské bílé, nové; 00004 10,50 Ponožky pánské tmavé; 00016 380,00 Slepíčí peří na polštáře. At the bottom are two buttons: 'Končit' (End) and 'Tisk' (Print)." data-bbox="113 472 635 705"/>

CZBOZI	CENAJ	NAZZBO
00017	130,00	Hrábě dřevěné, kolíky plastové
00018	56,00	Husí sádlo v konzervě
00013	360,00	Míč 
00012	45,00	Pálka pingpongová
00006	10,55	Ponožky pánské bílé, nové
00004	10,50	Ponožky pánské tmavé
00016	380,00	Slepíčí peří na polštáře

Stejně jako v předchozím dotazu se nyní uživatel může spokojit s výsledkem a ukončit dotaz, nebo může výsledek vytisknout na tiskárně.

Tento skript, UTF_8_MARKERS, který ovšem uživatel nevidí, má v příkazu SELECT definovaná místa proměnných hodnot označená otazníky.

```
--Výběr podle parametrů zadaných uživatelem
```

```
--;? 1; CHAR ; Zboží <= : ; 00020
--;? 2; DEC  ; Cena mezi ; 10.00
--;? 3;      ;          ; 500.00
--;? 4;      ; Název mezi; B
--;? 5;      ;          ; T
```

```
SELECT CZBOZI, CENAJ, NAZZBO FROM CENY4
WHERE CZBOZI <= ?
AND CENAJ BETWEEN ? AND ?
AND NAZZBO BETWEEN ? AND ?
ORDER BY NAZZBO ASC
```

Komentářové řádky (vyznačené kurzívou) začínají čtyřznakem *--;?* v prvním sloupci a definují čtveřici údajů oddělených středníkem. Umožňují sestavit výzvu uživateli k zadání proměnných hodnot na místa otazníků.

První údaj je *pořadové číslo*, které musí přesně odpovídat pořadí daného otazníku v příkazu. Může, ale nemusí obsahovat vedoucí nuly.

Druhý údaj je typ dat pro sloupec v SQL (v tomto příkladu je to CHAR a DEC). Slouží pro vlastní kontrolu správnosti zadané hodnoty a pro dokumentaci. Údaj může být i prázdný, ale nesmí chybět jeho koncový středník.

Třetí údaj je doprovodný text k zadávané hodnotě, zde Zboží <= :, Cena mezi a Název mezi. Také tento údaj může být prázdný, ale s koncovým středníkem. Mezery se promítají do doprovodného textu.

Čtvrtý údaj je předvolená hodnota odpovídající typu dat, zde je to znakově kódované číslo zboží 00020 , začáteční písmeno názvu zboží, a destinná čísla cen. Tento údaj může, ale nemusí končit středníkem.

Nedotazový skript

Jako ilustrace je uveden skript UTF_8_CREATE_INSERT.sql.

```
--Vytvořit tabulku CENY4 s kódováním textu UTF-8
```

```
DELETE FROM VZTOOL.CENY4;
```

```
CREATE TABLE VZTOOL.CENY4  
( CZBOZI CHAR(5) PRIMARY KEY,  
  CENAJ DEC(12, 2),  
  NAZZBO CHAR(50) CCSID 1208,  
  RAZITKO TIMESTAMP  
);
```

```
SET SCHEMA DEFAULT -- *LIBL
```

Každý příkaz kromě posledního je ukončen středníkem. U příkazu CREATE je jméno tabulky CENY4 kvalifikováno jménem schematu VZTOOL.



Nejdříve se ukáže okno s chybným příkazem. Po stisku tlačítka *Končit* pokračuje v dalším výpočtu. Jestliže potom nedojde k chybě, provedou se další příkazy a nakonec se zobrazí okno se souhrnem všech příkazů a zpráv.



Jako příklad ještě následuje vložení dat do tabulky skriptem UTF_8_INSERT.

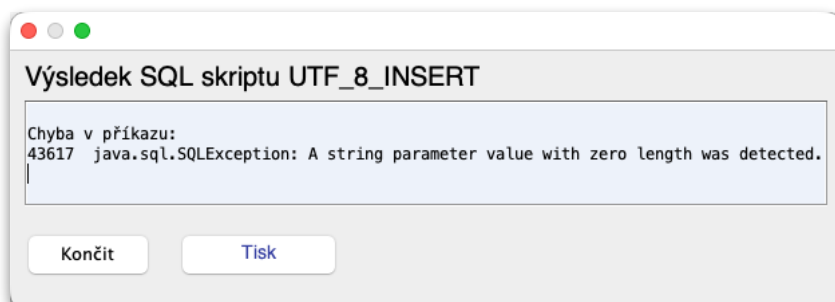
```
--Vložení dat do tabulky CENY4
```

```
SET SCHEMA DEFAULT; -- *LIBL
```

```
DELETE FROM CENY4;
```

```
INSERT INTO CENY4 values ('00001', 8,99, 'PIŠKOTY OPAVIA', '2016-01-18 17:34:55');
INSERT INTO CENY4 values ('00002', 459,00, 'Zubní pasta Kalodont', '2016-01-18 17:34:55');
INSERT INTO CENY4 values ('00003', 1,25, 'Prádelní šňůra', '2016-01-18 17:34:55');
INSERT INTO CENY4 values ('00004', 10,50, 'Ponožky pánské tmavé', '2016-01-18 17:34:55');
INSERT INTO CENY4 values ('00005', 120,00, 'Tričko bílé', '2016-01-18 17:34:55');
INSERT INTO CENY4 values ('00006', 10,55, 'Ponožky pánské bílé, nové', '2016-01-18 17:34:55');
INSERT INTO CENY4 values ('00007', 1510,00, 'Kalhoty manšestrové', '2016-01-18 17:34:55');
INSERT INTO CENY4 values ('00008', 1700,00, 'Kalhoty džínové', '2016-01-18 17:34:55');
INSERT INTO CENY4 values ('00009', 250,00, 'Whisky Balantine', '2016-01-18 17:34:55');
INSERT INTO CENY4 values ('00010', 6500,00, 'Koňak Gruzínský', '2016-01-18 17:34:55');
INSERT INTO CENY4 values ('00011', 159,00, 'Taška sportovní', '2016-01-18 17:34:55');
INSERT INTO CENY4 values ('00012', 45,00, 'Pálka pingpongová', '2016-01-18 17:34:55');
INSERT INTO CENY4 values ('00013', 360,00, 'Míč', '2016-01-18 17:34:55');
INSERT INTO CENY4 values ('00014', 3500,00, 'Sako tvídové, nadměr', '2016-01-18 17:34:55');
INSERT INTO CENY4 values ('00015', 12500,00, 'Motorové koště', '2016-01-18 17:34:55');
INSERT INTO CENY4 values ('00016', 380,00, 'Slepičí peří na polštáře', '2016-01-18 17:34:55');
INSERT INTO CENY4 values ('00017', 130,00, 'Hrábě dřevěné, kolíky plastové', '2016-01-18 17:34:55');
INSERT INTO CENY4 values ('00018', 56,00, 'Husí sádlo v konzervě', '2016-01-18 17:34:55');
INSERT INTO CENY4 values ('00452', 874,00, 'Tiskárna barevná laserová', '2016-01-18 17:34:55');
```

Spuštěním skriptu vznikne okno s chybovou zprávou, která je nejspíš chybou v soustavě *IBM i Toolbox for Java*.



Stiskem tlačítka *Končit* dostaneme souhrn výsledků.

Výsledek SQL skriptu UTF_8_INSERT

—Vložení dat do tabulky CENY4

SET SCHEMA DEFAULT -- *LIBL
Příkaz se provedl.

DELETE FROM CENY4
Příkaz se provedl.

INSERT INTO CENY4 values ('00001', 8,99, 'PIŠKOTY OPAVIA', '2016-01-18 17:34:55')
Příkaz se provedl.
INSERT INTO CENY4 values ('00002', 459,00, 'Zubní pasta Kalodont', '2016-01-18 17:34:55')
Příkaz se provedl.
INSERT INTO CENY4 values ('00003', 1,25, 'Prádelní šňůra', '2016-01-18 17:34:55')
Příkaz se provedl.
INSERT INTO CENY4 values ('00004', 10,50, 'Ponožky pánské tmavé', '2016-01-18 17:34:55')
Příkaz se provedl.
INSERT INTO CENY4 values ('00005', 120,00, 'Tričko bílé', '2016-01-18 17:34:55')
Příkaz se provedl.
INSERT INTO CENY4 values ('00006', 10,55, 'Ponožky pánské bílé, nové', '2016-01-18 17:34:55')
Příkaz se provedl.
INSERT INTO CENY4 values ('00007', 1510,00, 'Kalhoty manšestrové', '2016-01-18 17:34:55')
Příkaz se provedl.
INSERT INTO CENY4 values ('00008', 1700,00, 'Kalhoty džínové', '2016-01-18 17:34:55')
Příkaz se provedl.
INSERT INTO CENY4 values ('00009', 250,00, 'Whisky Balantine', '2016-01-18 17:34:55')
Příkaz se provedl.
INSERT INTO CENY4 values ('00010', 6500,00, 'Koňak Gruzinský', '2016-01-18 17:34:55')
Příkaz se provedl.
INSERT INTO CENY4 values ('00011', 159,00, 'Taška sportovní', '2016-01-18 17:34:55')
Příkaz se provedl.
INSERT INTO CENY4 values ('00012', 45,00, 'Pálka pingpongová', '2016-01-18 17:34:55')
Příkaz se provedl.
INSERT INTO CENY4 values ('00013', 360,00, 'Míč 🏐', '2016-01-18 17:34:55')
Příkaz se provedl.
INSERT INTO CENY4 values ('00014', 3500,00, 'Sako tvídové, nadměr', '2016-01-18 17:34:55')
Příkaz se provedl.
INSERT INTO CENY4 values ('00015', 12500,00, 'Motorové koště', '2016-01-18 17:34:55')
Příkaz se provedl.
INSERT INTO CENY4 values ('00016', 380,00, 'Slepičí peří na polštáře', '2016-01-18 17:34:55')
Příkaz se provedl.
INSERT INTO CENY4 values ('00017', 130,00, 'Hrábě dřevěné, kolíky plastové', '2016-01-18 17:34:55')
Příkaz se provedl.
INSERT INTO CENY4 values ('00018', 56,00, 'Husí sádlo v konzervě', '2016-01-18 17:34:55')
Příkaz se provedl.
INSERT INTO CENY4 values ('00452', 874,00, 'Tiskárna barevná laserová', '2016-01-18 17:34:55')
Příkaz se provedl.

Chyba v příkazu:
43617 java.sql.SQLException: A string parameter value with zero length was detected.

Končit

Tisk

Pravidla pro tvorbu skriptů

Obsahem skriptu je především text příkazu SQL nebo několika příkazů oddělených středníkem. Poslední příkaz končí posledním řádkem. Neukončuje se středníkem (byl by to prázdný příkaz). Kromě příkazů může skript obsahovat komentáře, a to jednoduché a složené.

Jednoduchý komentář začíná dvěma souvislými pomlčkami (znaky `--`) a končí na konci textového řádku. Jednoduchý komentář je použit k zadávání speciálních parametrických údajů pro program, který je vyhodnocuje před provedením nebo po provedení skriptu.

Složený komentář začíná znaky `/*` a končí znaky `*/`, podobně jako v jiných programovacích jazycích. Může se vyskytovat všude, kde je v příkazu předepsána mezera. Složený komentář se k zadávání parametrů aplikace nepoužívá.

První řádek skriptu obsahující komentář začínající znaky `--` od 1. sloupce, se považuje za *popis skriptu*. Znaky zapsané za dvojznakem `--` se objeví v seznamu skriptů jako popis.

Důležitou složkou skriptů jsou *speciální komentářové řádky*, které začínají v 1. sloupci řádku speciálními čtyřznaky a představují definice následujících parametrů.

`--;?` proměnná hodnota parametru (v příkazu SQL označená otazníkem),
`--;t` titulní hlavička,
`--;H` sloupcový hlavičkový řádek (hlavičky sloupců),
`--;T` svislé a vodorovné členění řádků, symbol pro prázdné sloupce (null),
`--;D` masky pro výstupní formát dekadických čísel typu DECIMAL a NUMERIC,
`--;O` seznam sloupců, které se mají vynechat z výstupu,
`--;L` skupinová úroveň, její úvodní text a jméno skupinového sloupce,
`--;S` jméno sumarizovaného sloupce a seznam typů souhrnů (S, A, M, m, C),
`--;s` seznam úvodních textů pro sumární řádky (místo standardních SUM, AVG, MAX, MIN, COUNT),
`--;P` parametry pro tisk.

Čtyřznaky byly voleny tak, aby údaje specifikované za nimi mohly obsahovat libovolné znaky kromě středníku, který údaje odděluje.

Definice proměnných hodnot parametrů (parameter markers).

Otazníky v SQL příkazu označují *značky parametrů* (parameter markers). Místo nich se před provedením příkazu dosadí určené hodnoty. Dosazování hodnot je umožněno prostřednictvím komentářových definičních řádků. Počet a pořadí definičních řádků určuje počet a pořadí vstupních polí v okně, kde uživatel zadává hodnoty. Definiční řádek začíná čtyřznakovým symbolem `--;?` v 1. pozici a určuje

- pořadové číslo parametru (markeru) v SQL příkazu,
- typ dat v parametru,
- text k vysvětlení významu parametru,
- předem zvolená hodnotá parametru.

Obecný tvar definice je následující. Hranaté závorky označují nepovinný údaj.

```
--;? pořadové-číslo; [typ-dat]; [vysvětlení]; [předvolená-hodnota] [;]
```

Definičních řádků musí být přesně tolik, kolik je v příkazu SQL otazníků. Každý řádek obsahuje za symbolem `--;?` přesně čtyři údaje oddělené středníkem. Vedoucí a koncové mezery v údajích se ignorují.

Poznámka: Je-li ve skriptu více SQL příkazů než jeden, mohou být proměnné parametry zadány *jen v jednom SQL příkazu*.

Údaje v definičním řádku

Pořadové číslo

První údaj je povinné pořadové číslo, které musí přesně odpovídat pořadí daného otazníku v SQL příkazu. Je to 1 pro první otazník, 2 pro druhý otazník atd. Neurčuje pořadí vstupních polí v okně, to je dáno pořadím definičních řádků ve skriptu.

Typ dat

Druhý údaj je nepovinný a označuje některý z následujících datových typů.

DEC
DECIMAL
NUMERIC
INT
INTEGER
BIGINT
DATE
TIME
TIMESTAMP

Vyjmenované typy slouží pro předběžnou kontrolu správnosti hodnoty zadané do vstupního pole, ale také pro dokumentaci. Je-li takový typ zadán, provádí se kontrola hodnoty ještě *před spuštěním SQL příkazu*. Není-li zadán nebo je zadán jiný typ anebo libovolný text, provede se kontrola hodnoty podle typu *až v době provádění SQL příkazu*. Údaj může být prázdný, ale nesmí chybět středník.

Vysvětlující text

Třetí údaj je nepovinný vysvětlující text k zadávané hodnotě parametru, např. `Zboží <= :.` Také tento údaj může být prázdný nebo plný mezer, a musí být ukončen středníkem. Okrajové mezery se neuplatní.

Předvolená hodnota

Čtvrtý údaj je předvolená hodnota odpovídající typu dat, např. cena `5.50`. Tento údaj může být prázdný a může, ale nemusí končit středníkem.

Údaje typu DEC, DECIMAL, NUMERIC mohou obsahovat jen číslice, desetinnou *tečku* a znaménko před číslem. Desetinná čárka se při této kontrole považuje za chybu.

Údaje typu INT, INTEGER, BIGINT mohou obsahovat jen číslice a znaménko před číslem.

Údaje typu DATE musí být zadány ve formátu ISO, tedy YYYY-MM-DD, např. 2014-02-15.

Údaje typu TIME musí být zadány ve formátu ISO, tedy HH:MM:SS, např. 19:31:05.

Údaje typu TIMESTAMP (časové razítko) mohou být zadány ve formátu ISO, tedy

YYYY-MM-DD HH:MM:SS.MMMMMM, např. 2000-04-05 23:59:59.999999.

Jeho délka je 26 znaků. Mezi datem a časem je mezera. Vedoucí nuly lze vynechat z měsíce, dne, hodiny, minuty a sekundy u tvaru s oddělovači. Koncové nuly mikrosekund mohou být částečně nebo zcela vynechány. Připouští se i tvar s koncovkou 24:00:00.000000 (půlnoc).

Je-li typ sloupce BINARY nebo VARBINARY, je třeba předvolenou hodnotu zapsat ve tvaru s *hexadecimálními* znaky (0123456789abcdefABCDEF). Jednomu bajtu odpovídají dva hexadecimální znaky.

Řádky definic mohou být zapsány v libovolném pořadí; důležité je jejich pořadové číslo.

Příklad definic:

```
--;?01 ; ; Cena od: ; 5.50;  
--;? 02;DEC; Cena do:; 23000  
--;? 03; DATE; Od data:;2014-01-29;  
--;? 04; ; Binární data - dva bajty: ; 0F9C;
```

Zadání hodnot parametrů při spouštění skriptu

V okně výzvy k zadání proměnných parametrů se kontrolují jak data z definic, tak i data, která zadá uživatel ve vstupních polích. Kontrolují se, zda jsou v souladu se *zadaným* typem (v definici `--;?`), a to ještě předtím, než se spustí provádění skriptu (SQL příkazu).

- U typu DEC, DECIMAL, NUMERIC se kontroluje správnost zadaného čísla. Číslo musí obsahovat jen číslice, znaménko (plus nebo minus) vlevo a desetinnou *tečku*. Desetinná *čárka* se považuje za chybu a program ji ihned ji ohlásí.
Není-li typ takto přesně zadán, čárka se v *české* lokalizaci nepovažuje za chybu a SQL ji přijme. V *anglické* lokalizaci se desetinná čárka považuje vždy za chybu.
- Typ DATE se kontroluje na správnost zápisu podle ISO.
- Typ TIME se kontroluje na správnost zápisu podle ISO.
- Typ TIMESTAMP se kontroluje na správnost zápisu podle ISO.

Jestliže ve specifikačním řádku typ chybí nebo je zadán jiný text, závada v zadané hodnotě se projeví jako chyba až v době provádění SQL příkazu. Zpráva SQL o chybě bude patrná z výsledku, který se po provedení skriptu zobrazí.

Poznámka 1: Je-li typ sloupce BINARY nebo VARBINARY, je třeba hodnotu zapsat ve tvaru s hexadecimálními znaky (0123456789abcdefABCDEF). Jednomu bajtu odpovídají dva hexadecimální znaky. Hodnota se nekontroluje na správnost. Za chybný znak se dosadí 0.

Poznámka 2: Výsledky dotazu ve sloupci typu BINARY nebo VARBINARY jsou také zobrazeny v hexadecimálním vyjádření.

Příklad zadání parametrů

Skript se jménem UTF_8_QUERY definuje dva parametry.

```
---Jednoduchý dotaz CENY4 v kódování UTF-8
--;? 1 ; NUMERIC ; Jednotková cena větší nebo rovná: ; 1.00;
--;? 2 ; TIMESTAMP ; Časové razítko:; 2016-01-18 17:34:55;
```

```
select CZBOZI, CENAJ, NAZZBO, RAZITKO from CENY4
where CENAJ >= ?
and RAZITKO >= ?
order by CZBOZI
```

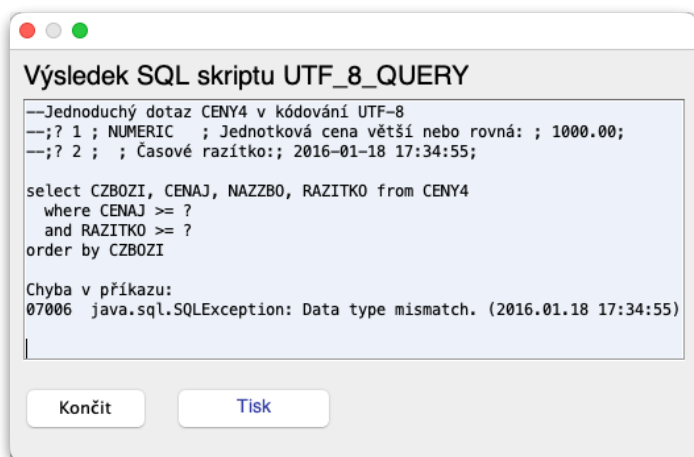
Po spuštění skriptu se zobrazí výzva k zadání hodnot a ke spuštění tlačítkem *Potvrdit*. Změníme-li předtím desetinnou tečku na čárku, zobrazí se zpráva o chybě.

Opravíme-li desetinnou čárku zpátky na tečku, ale v časovém razítku změníme pomlčky na tečky, zobrazí se zpráva o druhé chybě.

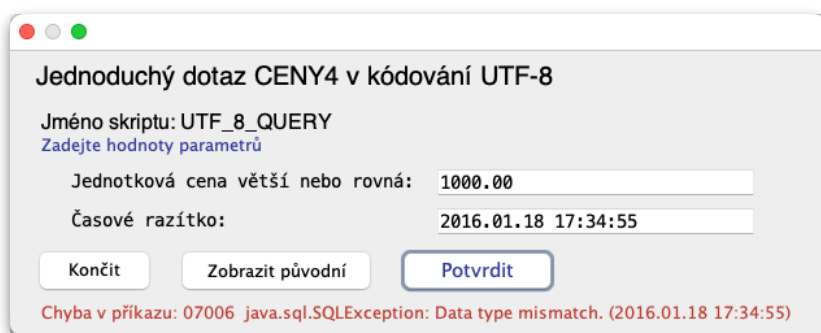
Opravíme-li i druhou chybu, dostaneme výsledek dotazu.

CZBOZI	CENAJ	NAZZBO	RAZITKO
00007	1510,00	Kalhoty manšestrové	2016-01-18 17:34:55.0
00008	1700,00	Kalhoty džínové	2016-01-18 17:34:55.0
00010	6500,00	Koňak Gruzínský	2016-01-18 17:34:55.0
00014	3500,00	Sako tvídové, nadměr	2016-01-18 17:34:55.0
00015	12500,00	Motorové koště	2016-01-18 17:34:55.0

Jestliže v druhé definici vynecháme typ `TIMESTAMP` a uděláme v něm předchozí chybu (tečky v časovém razítku), výsledek bude okno s kopií skriptu a s chybovou zprávou z SQL.



Po uzavření výsledkového okna nebo stiskem tlačítka *Končit* se opět zobrazí výzva k zadání parametrů se stejnou chybovou zprávou.



Tlačítko *Zobrazit původní* zobrazí okno v původní podobě.



Definice titulních hlaviček

Pro dotazový skript můžeme definovat libovolný počet titulních hlavičkových řádků. Ty se uplatní v pořadí v jakém jsou napsány ve skriptu. Definice titulního hlavičkového řádku začíná v 1. sloupci čtyřznakovým symbolem **--;t.** za nímž může následovat libovolný text.

Titulní hlavičkový řádek může obsahovat libovolný text a v něm případně označení proměnných tvaru

&sloupec

kde *sloupec* je jméno sloupce, za nímž následuje mezera nebo konec řádku. Za tuto proměnnou se dosadí hodnota stejnojmenného sloupce z *prvního řádku* výsledné tabulky, i když je tento sloupec pojmenován v parametrech jako vynechaný (viz dále).

Titulní hlavičkové řádky se vypisují před sloupcovými hlavičkami a tisknou se jen na první stránce.

Skript ZBOZI_VYNECHANE ukazuje titulní řádky a vynechané sloupce. V příkladu proměnné **&ZAVOD** a **&CELKEM** odpovídají sloupcům ZAVOD a CELKEM.

--Zboží podle skladů a závodů - vynechané sloupce a proměnné v titulcích

```
--;t   Závod z prvního řádku: &ZAVOD , Celková cena z prvního řádku: &CELKEM ,
--;t   ~~~~~
--;t   Celková cena znovu: &Celkem
--;t
--;HŽav;Skl;Č.zboží;Název zboží;Cena/j.;Množství;   Cena celkem;
--;H===;===;=====;=====;=====;=====;   =====;
--;H
/* Dva vynechané sloupce */
--;O závod ;
--;O celkem
```

```
SELECT ZAVOD, SKLAD, S.CZBOZI, NAZZBO,
       DEC(CENAJ, 12, 2) AS CJ,   MNOZSTVI,
       DECIMAL (CENAJ * MNOZSTVI, 12, 2 ) AS CELKEM
FROM STAVY AS S
JOIN CENY4 AS C ON C.CZBOZI = S.CZBOZI
WHERE ZAVOD = '02' AND S.CZBOZI <> '00002'
ORDER BY SKLAD desc, S.CZBOZI
```

Výsledek je na následujícím obrázku.

Výsledek SQL skriptu ZBOZI_VYNECHANE

Zboží podle skladů a závodů - vynechané sloupce a proměnné v titulcích

čtvrtek 11. prosince 2025 11:45:40

Závod z prvního řádku: 02, Celková cena z prvního řádku: 500,00,
~~~~~  
Celková cena znovu: 500,00

| Skl | Č.zboží | Název zboží               | Cena/j. | Množství |
|-----|---------|---------------------------|---------|----------|
| === | =====   | =====                     | =====   | =====    |
| 02  | 00009   | Whisky Balantine          | 250,00  | 2,00     |
| 02  | 00011   | Taška sportovní           | 159,00  | 2,00     |
| 02  | 00014   | Sako tvídové, nadměr      | 3500,00 | 2,00     |
| 02  | 00018   | Husí sádlo v konzervě     | 56,00   | 2,00     |
| 01  | 00005   | Tričko bílé               | 120,00  | 2,00     |
| 01  | 00006   | Ponožky pánské bílé, nové | 10,55   | 2,00     |
| 01  | 00008   | Kalhoty džínové           | 1700,00 | 2,00     |

## Definice sloupcových hlaviček

### Vlastní hlavičky

Pro dotazový příkaz SELECT můžeme definovat libovolný počet hlavičkových řádků. Ty se uplatní v pořadí v jakém jsou napsány ve skriptu. Hlavičkový řádek začíná v 1. sloupci čtyřznakovým symbolem **--;H** a za ním následuje potřebný počet textů (sloupcových nadpisů) oddělených středníkem.

```
--;H [ [text-1] [; text-2] [; ...] [; text-n] [;] ]
```

Do výstupu dotazu se berou texty včetně vedoucích a koncových mezer. Lze tak například zarovnávat víceřádkové nadpisy sloupců nebo zvětšovat šířku zobrazených sloupců dat.

Ve skriptu ZBOZI\_VYNECHANE je definice hlaviček

```
--;HZav;Sk1;Č.zboží;Název zboží;Cena/j.;Množství;   Cena celkem;  
--;H====;====;=====;=====;=====;=====;  
--;H
```

která vytvoří ve výsledku dotazu tři řádky hlaviček, z nichž třetí je prázdný:

| Sk1 | Č.zboží | Název zboží      | Cena/j. | Množství |
|-----|---------|------------------|---------|----------|
| === | =====   | =====            | =====   | =====    |
| 02  | 00009   | Whisky Balantine | 250,00  | 2,00     |
| ... |         |                  |         |          |

### Standardní hlavičky

Není-li zadán žádný řádek definice hlaviček, vypíše se standardní hlavička s *názvy sloupců* definovaných v tabulce, která je výsledkem dotazu. Standardní hlavička může být původní, ale také přejmenovaný název sloupce. Ten může být jednoduchý (ve velkých písmenech) nebo vlastní, zapsaný v uvozovkách.

Následující skript STANDARDNI\_HLAVICKY obsahuje frázi GROUP BY ROLLUP, která určuje seskupování výsledných řádků podle skladu a závodu. V něm jsou všechny hlavičky standardní (tedy neurčené definicí hlaviček). Součtové řádky (označené červeně) mají ve sloupci SKLAD pomlčku, protože má hodnotu NULL. Podobně ve sloupci ZAVOD. Hodnota NULL je vyjádřena pomlčkou, protože je právě zadána v parametrech aplikace.

```
SELECT S.ZAVOD, S.SKLAD,  
       DECIMAL( SUM(C.CENAJ*S.MNOZSTVI), 9, 2 ) AS "Součet",  
       INT( AVG( C.CENAJ*S.MNOZSTVI ) ) AS "Průměr",  
       INT( MIN( C.CENAJ*S.MNOZSTVI ) ) AS MIN,  
       INT( MAX( C.CENAJ*S.MNOZSTVI ) ) AS MAXIMUM  
FROM STAVY AS S  
JOIN CENY AS C ON S.CZBOZI = C.CZBOZI  
GROUP BY ROLLUP( S.ZAVOD, S.SKLAD )  
ORDER BY S.ZAVOD, S.SKLAD
```

| ZAVOD | SKLAD | Součet   | Průměr | MIN | MAXIMUM |
|-------|-------|----------|--------|-----|---------|
| 01    | 01    | 60158,20 | 20052  | 459 | 45500   |
| 01    | 02    | 19793,75 | 6597   | 23  | 19500   |
| 01    | 03    | 1,25     | 1      | 1   | 1       |
| 01    | -     | 79953,20 | 11421  | 1   | 45500   |
| 02    | 01    | 3661,10  | 1220   | 21  | 3400    |
| 02    | 02    | 7930,00  | 1982   | 112 | 7000    |
| 02    | -     | 11591,10 | 1655   | 21  | 7000    |
| -     | -     | 91544,30 | 6538   | 1   | 45500   |

## Definice vertikálního a horizontálního členění

Definiční řádek skupin může být jen jeden. Je-li jich více, uplatní se jen první. Začíná v 1. sloupci řádku čtyřznakovým symbolem **--;T** a za ním následují údaje oddělené středníkem definující

- počet prázdných řádků *před* řádkem nebo skupinou řádků *n* nebo **Bn**,
- počet prázdných řádků *za* řádkem nebo skupinou řádků *n* nebo **An**,
- symbol nahrazující prázdnou (NULL) hodnotu sloupce,
- počet mezer oddělujících sloupce ve výstupním řádku **Sn**,
- jedno nebo více jmen sloupců (oddělených středníkem) pro potlačení duplicitních hodnot.

```
--;T [ [počet-před] [;počet-za] [;symbol-pro-null] [;mezery-mez-sloupce]
      [;jméno-sloupce] ... ] [;] ]
```

Parametr *počet-před* má tvar **Bn** nebo **n**, kde *n* je celé číslo. Parametr *počet-za* má tvar **An** nebo **n**. Parametr *mezery-mez-sloupce* má tvar **Sn** nebo **n**. První tři parametry mohou být prázdné. Je-li některý z prvních dvou prázdný nebo chybný, dosadí se číslo *nula*. Je-li parametr *symbol-pro-null* prázdný, dosadí se hodnota z parametrů aplikace. Je-li parametr *počet-mezer* prázdný nebo chybný, dosadí se počet oddělujících mezer z parametrů aplikace.

Pátý a další parametr určuje *jméno sloupce*, u nějž se sleduje opakování stejné hodnoty ve skupně následujících řádků. Hodnota takového sloupce bude zapsána jen v prvním řádku takové skupiny, v ostatních řádcích budou mezery. První jméno platí pro nejvyšší skupinovou úroveň, druhé jméno pro druhou úroveň, atd.

Poznámka: Prázdné řádky se zařazují do výsledného dotazu před nebo za řádky nebo skupiny řádků, které jsou výsledkem sumarizace podle skupinových úrovní (viz dále). Prázdné řádky se zařazují také před nebo za jednotlivé řádky, když obsahují prázdné hodnoty (NULL) v některém sloupci, není-li zadána sumarizace podle skupinových úrovní.

Skript ZBOZI\_PREHLED ilustruje úpravu hlaviček, skupinových úrovní, součtových skupin a sloupců, u nichž se vypíše jen hodnota v prvním řádku skupinové úrovně.

```
--Zboží podle skladů a závodů
--;HZav;Skl;Č.zboží;Název zboží;Cena/j.;Množství;   Cena celkem;
--;H===;===;=====;=====;=====;=====;
--;H

--;T B1; A1; ; S5; ZAVOD ; SKLAD

SELECT ZAVOD, SKLAD, S.CZBOZI,
       SUBSTR(NAZZBO, 1, 15) AS NAZ,
       DEC(CENAJ, 12, 2) AS CJ, MNOZSTVI,
       DECIMAL ( SUM(CENAJ * MNOZSTVI), 12, 5 ) AS CELKEM
FROM STAVY AS S
JOIN CENY4 AS C ON S.CZBOZI = C.CZBOZI
GROUP BY ROLLUP( ZAVOD, ( SKLAD, S.CZBOZI, NAZZBO, CENAJ, MNOZSTVI ) )
ORDER BY ZAVOD, SKLAD, S.CZBOZI
```

Výsledek je vidět na obrázku:

## Výsledek SQL skriptu ZBOZI\_PREHLED

Zboží podle skladů a závodů

čtvrtek 11. prosince 2025 13:51:11

| Zav | Skl | Č.zboží | Název zboží     | Cena/j. | Množství | Cena celkem |
|-----|-----|---------|-----------------|---------|----------|-------------|
| 01  | 01  | 00001   | PIŠKOTY OPAVIA  | 8,99    | 1579,44  | 14199,20    |
|     |     | 00002   | Zubní pasta Ka  | 459,00  | 1,00     | 459,00      |
|     |     | 00010   | Koňak Gruzíns   | 6500,00 | 7,00     | 45500,00    |
|     | 02  | 00003   | Prádelní šň     | 1,25    | 19,00    | 23,75       |
|     |     | 00009   | Whisky Balantin | 250,00  | 1,08     | 270,00      |
|     |     | 00010   | Koňak Gruzíns   | 6500,00 | 3,00     | 19500,00    |
|     | 03  | 00003   | Prádelní šň     | 1,25    | 1,00     | 1,25        |
|     |     |         |                 |         |          | 79953,20    |
| 02  | 01  | 00005   | Tričko bílé     | 120,00  | 2,00     | 240,00      |
|     |     | 00006   | Ponožky pánsk   | 10,55   | 2,00     | 21,10       |
|     |     | 00008   | Kalhoty džino   | 1700,00 | 2,00     | 3400,00     |
|     | 02  | 00009   | Whisky Balantin | 250,00  | 2,00     | 500,00      |
|     |     | 00011   | Taška sportovn  | 159,00  | 2,00     | 318,00      |
|     |     | 00014   | Sako tvídové,   | 3500,00 | 2,00     | 7000,00     |
|     |     | 00018   | Husí sádlo v    | 56,00   | 2,00     | 112,00      |
|     |     |         |                 |         |          | 11591,10    |
|     |     |         |                 |         |          | 91544,30    |

Končit

Tisk

Zde jsou skupiny závodů označeny jen v prvním řádku skupiny. Skupiny skladů rovněž (i když se za ně nesčítá). Prázdné hodnoty sloupců jsou nahrazeny mezerou a řádkování součtů je B1 “jeden prázdný před” a A1 “jeden prázdný po” součtu. Sčítá se za ZAVOD a CELKEM.

Poznámka: Čtvrtá hodnota S5 značí počet mezer mezi sloupci. Tento parametr musí být zadán (alespoň koncovým středníkem), jestliže jsou zadána jména sloupců pro potlačení duplicit.



## Úprava čísel pro výstup

Dekadická čísla typu DEC, DECIMAL, NUMERIC jsou ve výsledku dotazu upravena formátem. Formát je buď *standardní* pro danou lokalizaci nebo předepsaný *maskou*.

Poznámka: Maska se neuplatňuje pro celočíselné typy INT, INTEGER, BIGINT.

### Symbole pro tvorbu masek

V následující tabulce jsou uvedeny znaky tvořící masku, jejich umístění a význam.

| Symbol         | Umístění v masce       | Podléhá lokalizaci | Význam v české lokalizaci                                                                                    |
|----------------|------------------------|--------------------|--------------------------------------------------------------------------------------------------------------|
| 0              | v čísle                | ano                | Číslice                                                                                                      |
| #              | v čísle                | ano                | Číslice, přičemž nula je potlačena                                                                           |
| .              | v čísle                | ano                | Oddělovač desetinných míst - čárka                                                                           |
| -              | před číslem            | ano                | Přidá pohyblivé záporné znaménko. Neobrací číslo!                                                            |
| ,              | v čísle                | ano                | Oddělovač skupin číslic - mezera                                                                             |
| %              | před číslem, za číslem | ano                | Stonásobek označený jako procento                                                                            |
| ¤              | před číslem, za číslem | ne                 | Znak pro měnu nahrazený měnovým symbolem Kč. Je-li zdvojen, je nahrazen mezinárodním měnovým symbolem - CZK. |
| ‰              | před číslem, za číslem | ano                | Tisícínásobek označený jako promile                                                                          |
| libovolný text | před číslem            | ne                 | Pohyblivý text před číslem                                                                                   |
| libovolný text | za číslem nebo uvnitř  | ne                 | Pevný text za číslem                                                                                         |

Upozornění: Zápis znaků 0 mezi posloupnost znaků # a obráceně je chyba.

### Příklady masek

Následují příklady formátování čísel maskami v české lokalizaci.

| Číslo        | Maska     | Výstup          | Poznámka                                                                            |
|--------------|-----------|-----------------|-------------------------------------------------------------------------------------|
| 123456789.50 | #,###0.00 | 1 234 567,50    | Oddělí čtveřice číslic mezerou                                                      |
| 1234567.50   | ,##0.00 ¤ | 1 234 567,50 Kč | Oddělí trojice číslic mezerou a přidá český symbol měny                             |
| 1234567.50   | #0.00 ¤¤  | 1234567,50 CZK  | Přidá mezinárodní symbol lokální (zde české) měny                                   |
| 1234567.50   | #0.00 CZK | 1234567,50 CZK  | Přidá konstantu " CZK" za číslo. Za nebo před číslo lze přidávat libovolné symboly. |
| 1234567.50   | #0** .00  | 1234567,50**    | Hvězdičky se přidávají těsně za číslo                                               |

| Číslo           | Maska            | Výstup            | Poznámka                                                                                               |
|-----------------|------------------|-------------------|--------------------------------------------------------------------------------------------------------|
| -1234567.55     | #0.0             | -1234567,6        | Zaokrouhlí na jedno desetinné místo (polovinu na sudou číslici)                                        |
| -0.04           | #.00             | -,04              | Potlačí nulu před desetinnou čárkou                                                                    |
| 5.55            | 00000.000        | 00005,550         | Nepotlačuje vedoucí nuly a přidá nulu na konci                                                         |
| 0.56            | #0.00 %          | 56 %              | Vyjádření zlomku v procentech                                                                          |
| 0.007           | #0.0             | 7,0 ‰             | Vyjádření zlomku v promilích                                                                           |
| 12345<br>-12345 | -#0.00<br>-#0.00 | -12345<br>--12345 | Pohyblivé znaménko minus vlevo. Neobrací kladné číslo na záporné. K zápornému přidá ještě jeden minus. |

Jedna z obvyklých masek pro české prostředí je maska #0.00, která potlačuje levostranné nuly v celé části čísla (kromě poslední) a zachovává dvě desetinná místa včetně koncových nul.

## Zadání masek

Masku můžeme zadat jednak v *parametrech aplikace*, jednak v *komentářových řádcích* skriptu. Masky z parametrů aplikace (není-li prázdná) platí pro všechny číselné sloupce (i celočíselné), pro něž není individuálně zadána maska definičním komentářovým řádkem.

*Definiční řádek masky* začíná v 1. sloupci čtyřznakovým symbolem **--;D**, za ním následuje maska a za ní jméno sloupce. Může, ale nemusí být ukončen středníkem. Vedoucí a koncové mezery se odstraní. Definice masky má tvar

```
--;D maska ; jméno-sloupce [ ; ]
```

## Příklad na výstup čísel pomocí masky

Zadáme-li v předchozím skriptu následující definice, upraví se dekadická čísla podle masek.

```
--;D 0000.000; CJ  
--;D 00; MNOZSTVI  
--;D 00000000.00000; CELKEM
```

### Výsledek SQL skriptu ZBOZI

Zboží podle skladů a závodů

čtvrtek 11. prosince 2025 14:35:56

| ZAVOD | SKLAD | CZBOZI         | CJ       | MNOZSTVI | CELKEM         |
|-------|-------|----------------|----------|----------|----------------|
| 01    | 01    | 00001          | 0008,990 | 1579     | 00014199,20156 |
|       |       | 00002          | 0459,000 | 01       | 00000459,00000 |
|       |       | 00010          | 6500,000 | 07       | 00045500,00000 |
|       | 02    | 00003          | 0001,250 | 19       | 00000023,75000 |
|       |       | 00009          | 0250,000 | 01       | 00000270,00000 |
|       |       | 00010          | 6500,000 | 03       | 00019500,00000 |
|       | 03    | 00003          | 0001,250 | 01       | 00000001,25000 |
|       |       |                |          |          |                |
|       |       |                |          |          | 00079953,20156 |
|       |       |                |          |          | 00091544,30156 |
| 02    | 01    | 00005          | 0120,000 | 02       | 00000240,00000 |
|       |       | 00006          | 0010,550 | 02       | 00000021,10000 |
|       |       | 00008          | 1700,000 | 02       | 00003400,00000 |
|       | 02    | 00009          | 0250,000 | 02       | 00000500,00000 |
|       |       | 00011          | 0159,000 | 02       | 00000318,00000 |
|       |       | 00014          | 3500,000 | 02       | 00007000,00000 |
|       |       | 00018          | 0056,000 | 02       | 00000112,00000 |
|       |       | 00011591,10000 |          |          |                |

## Formát čísel bez masky

Není-li zadána žádná maska, tj. ani v parametrech aplikace, znamená to, že dekadické číslo se zobrazí *podle standardních lokálních pravidel*. Standardní pravidla pro českou lokalizaci jsou:

- Vedoucí nuly se potlačí, z nich se ponechá jen nula před desetinnou čárkou.
- Nulový zlomek se vynechá i s desetinnou čárkou.
- Koncové nuly v desetinných místech se potlačí.
- Trojice číslic v celé části se oddělují mezerou.

|          |          |                                                              |
|----------|----------|--------------------------------------------------------------|
| 12345.00 | 12 345   | Odstraní desetinná místa, doplní mezeru mezi stovky a tisíce |
| 12345.50 | 12 345,5 | Vynechá koncovou nulu z desetinných míst                     |
| 0.665    | 0,66     | Zaokrouhlí na sudou číslici                                  |

Je zřejmé, že tato možnost se v dotazech použije jen výjimečně.

Poznámka: Výstupní obrazy dekadických čísel v české lokalizaci obsahují desetinnou čárku, přestože v masce je zadána desetinná tečka.

Ve skriptu **FORMATY** nejsou žádné masky zadány.

```
--Součet hodnot za závod
SELECT ZAVOD,
       DECIMAL( SUM(C.CENAJ*S.MNOZSTVI), 9, 2 ) AS CELKEM
FROM STAVY AS S
JOIN CENY4 AS C ON S.CZBOZI = C.CZBOZI
GROUP BY ZAVOD
ORDER BY ZAVOD
```

Když necháme masku v parametrech aplikace *prázdnou*, a neuvedeme ani definici masky pro sloupec **SOUCET** ve skriptu, budou hodnoty sloupce obsahovat desetinná místa bez koncových nul (nebo žádná desetinná místa bez čárky) a mezeru mezi trojicemi číslic v celé části:

| ZAVOD | CELKEM   |
|-------|----------|
| 01    | 79 953,2 |
| 02    | 11 591,1 |

Zařadíme-li však do skriptu definici se symbolem  $\text{¤}$  pro měnu

```
--;D ,##0.00¤ ; SOUCET
```

dostaneme výsledek

| ZAVOD | CELKEM      |
|-------|-------------|
| 01    | 79 953,20Kč |
| 02    | 11 591,10Kč |

## Vynechání sloupců z výstupu

Definiční řádek začíná čtyřznakovým symbolem `--;O` v 1. pozici.

Definičních řádků může být libovolný počet. Každý řádek obsahuje za symbolem `--;O` jména sloupců oddělená středníkem. Údaje mohou obsahovat vedoucí a koncové mezery, které se ignorují. Obecný tvar definice je následující. Hranaté závorky označují nepovinný údaj.

```
--;O [ sloupec; sloupec; ... [;] ]
```

Vyjmenované sloupce se vynechají z výstupu a tisku, a to jak hodnoty, tak sloupcové hlavičky.

### Příklad na vynechání sloupců a tisk titulních hlaviček

V následujícím příkladu jsou zadány titulní hlavičkové řádky s proměnnými a k tomu zadání pro vynechávané sloupce. Jak je vidět, řádků `--;O` může být i více. V titulních řádcích jsou zadány proměnné jako jména sloupců s předřazeným znakem `&` (ampersand), které shodou okolností odkazují na vynechané sloupce. Provedením následujícího skriptu se do titulních hlaviček dosadí hodnoty dotyčných sloupců z *prvního* řádku výsledku dotazu.

```
--Zboží podle skladů a závodů - vynechané sloupce a proměnné v titulcích
```

```
--;t   Závod z prvního řádku: &ZAVOD , Celková cena z prvního řádku: &CELKEM ,
--;t   -----
--;t   Celková cena znovu: &Celkem
--;t
```

```
--;HZav;Skl;Č.zboží;Název zboží;Cena/j.;Množství;   Cena celkem;
--;H===;===;=====;=====;=====;=====;=====;
--;H
```

```
/* Tři mezery mezi sloupci */
--;T   ;   ;   ; 3
```

```
/* Dva vynechané sloupce */
--;O   závod ;
--;O   celkem
```

```
/* Parametry pro tisk */
--;P a4; fs9; Landscape; lm5; rm30; tm5; bm5;
```

```
SELECT S.ZAVOD, S.SKLD, S.CZBOZI, C.NAZZBO,
       DEC(C.CENAJ, 12, 2) AS CJ,   S.MNOZSTVI,
       DECIMAL (C.CENAJ*S.MNOZSTVI, 12, 2 ) AS CELKEM
FROM STAVY AS S
JOIN CENY AS C ON C.CZBOZI = S.CZBOZI
WHERE S.ZAVOD = '02' AND S.CZBOZI <> '00002'
ORDER BY S.SKLD desc, S.CZBOZI
```

Vypíše se:

zboží podle skladů a závodů – vynechané sloupce a proměnné v titulcích

Čtvrtek, 3. prosince 2015 17:32:53

Závod z prvního řádku: 02, Celková cena z prvního řádku: 500,00,

Celková cena znovu: 500,00

| Sk1 | Č.zboží | Název zboží               | Cena/j. | Množství |
|-----|---------|---------------------------|---------|----------|
| === | =====   | =====                     | =====   | =====    |
| 02  | 00009   | Whisky Balantine          | 250,00  | 2,00     |
| 02  | 00011   | Taška sportovní           | 159,00  | 2,00     |
| 02  | 00014   | Sako tvídové, nadměr      | 3500,00 | 2,00     |
| 02  | 00018   | Husí sádlo v konzervě     | 56,00   | 2,00     |
| 01  | 00005   | Tričko bílé               | 120,00  | 2,00     |
| 01  | 00006   | Ponožky pánské bílé, nové | 10,55   | 2,00     |
| 01  | 00008   | Kalhoty džínové           | 1700,00 | 2,00     |

Kdybychom nezadali vynechávané sloupce a titulní hlavičky, dostali bychom tento výsledek:

Zboží podle skladů a závodů – vynechané sloupce a proměnné v titulcích

Čtvrtek, 3. prosince 2015 17:30:39

Závod z prvního řádku: 02, Celková cena z prvního řádku: 500,00,

Celková cena znovu: 500,00

| Zav | Sk1 | Č.zboží | Název zboží               | Cena/j. | Množství | Cena celkem |
|-----|-----|---------|---------------------------|---------|----------|-------------|
| === | === | =====   | =====                     | =====   | =====    | =====       |
| 02  | 02  | 00009   | Whisky Balantine          | 250,00  | 2,00     | 500,00      |
| 02  | 02  | 00011   | Taška sportovní           | 159,00  | 2,00     | 318,00      |
| 02  | 02  | 00014   | Sako tvídové, nadměr      | 3500,00 | 2,00     | 7000,00     |
| 02  | 02  | 00018   | Husí sádlo v konzervě     | 56,00   | 2,00     | 112,00      |
| 02  | 01  | 00005   | Tričko bílé               | 120,00  | 2,00     | 240,00      |
| 02  | 01  | 00006   | Ponožky pánské bílé, nové | 10,55   | 2,00     | 21,10       |
| 02  | 01  | 00008   | Kalhoty džínové           | 1700,00 | 2,00     | 3400,00     |

Do titulních hlaviček se tedy správně dostala hodnota 01 ze sloupce ZAVOD a hodnota 500,00 ze sloupce CELKEM.

Poznámka: Není nutné vázat titulní hlavičky s vynechávanými sloupci. Obojí lze používat samostatně.

## Parametry pro tisk

Definiční řádek parametrů pro tisk může být jen jeden. Je-li jich více, uplatní se jen první. Začíná v 1. sloupci řádku čtyřznakovým symbolem **--;P** a za ním následují údaje oddělené středníkem definující

- velikost papíru A4, A3 nebo LETTER (netestováno na papíře),
- velikost písma tvaru  $FSn$ , kde  $n$  je počet tiskových bodů, např.  $f s 9$ ,
- orientaci stránky PORTRAIT či P, nebo LANDSCAPE či L,
- levý okraj (left margin) tvaru  $LMn$ , kde  $n$  je počet milimetrů od okraje tisknutelné oblasti papíru,
- pravý okraj (right margin) tvaru  $RMn$ ,
- horní okraj (top margin) tvaru  $TMn$ ,
- spodní okraj (bottom margin) tvaru  $BMn$ ,

Definiční řádek má tvar

```
--;P velikost-papíru; FSn; orientace; LMn; RMn ; TMn ;BMn [ ; ]
```

Řádek může, ale nemusí být ukončen středníkem. Jednotlivé údaje mohou být vynechány, ale počet středníků do prvního neprázdného údaje musí být zachován.

- Je-li údaj velikosti papíru prázdný nebo je chybný, dosadí se A4.
- Je-li údaj velikosti písma prázdný nebo je nečíselný, dosadí se jeho hodnota z parametrů aplikace.
- Je-li údaj orientace stránky prázdný nebo je chybný, dosadí se PORTRAIT.
- Je-li údaj okraje prázdný nebo je chybný, dosadí se 10 mm.

## Sumarizace výsledků dotazu

Výsledek dotazu (result set) lze sumarizovat podle skupinových úrovní (level summarization). Počet úrovní není výslovně omezen. Skupinovou sumarizaci definuje soustava tří typů parametrických definic:

```
--;L skupinová úroveň, její úvodní text a jméno skupinového sloupce,  
--;S jméno sumarizovaného sloupce a seznam typů souhrnů (S, A, M, m, C),  
--;s seznam úvodních textů pro sumární řádky (místo standardních SUM, AVG, MAX, MIN, COUNT),
```

Řádků **--;L** je tolik, kolik skupinových úrovní se používá. Řádků **--;S** je tolik, kolik sloupců se podrobuje sumarizaci. Řádek **--;s** je nejvýše jeden. Řádky **--S;** a **--s;** mají smysl jen tehdy, jsou-li zadány nějaké řádky **--;L**.

### Definice skupinových úrovní

Definiční řádek začíná čtyřznakovým symbolem **--;L** v 1. pozici. Obecný tvar definice je následující.

```
--;L [level]; [úvodní-text-skupiny]; [sloupec-úrovně]; [NP] [;]
```

První údaj *level* je buď 0 nebo libovolný text, i prázdný. Pro názornost se volí čísla 1, 2, atd.

Druhý údaj *úvodní-text-skupiny* je libovolný text, který navíc může obsahovat *proměnné* tvaru **&sloupec**, za nímž následuje mezera nebo konec řádku. Na místo této proměnné se dosadí hodnota stejnojmenného sloupce z *posledního řádku* skupiny dané úrovně, a to i tehdy, když je tento sloupec pojmenován v parametrech **--;O** jako vynechaný.

Třetí údaj *jméno-sloupce* představuje sloupec definující skupinu dané úrovně. V úrovni 0 nemá význam.

Čtvrtý údaj NP znamená *novou stránku* (New Page). Uplatní se při tisku na papír. Jestliže se změní skupina označená tímto údajem, první řádek nové skupiny se vytiskne až na další stránce.

Zápis **--;L0** určuje koncovou úroveň (grand total či final total). Řádek s údajem L0 je nepovinný, ale je-li zadán, pak musí být uveden jako první. Jestliže řádek s L0 není zadán, koncová úroveň se nezpracuje. Údaj *level* může být u dalších (“nižších”) skupinových úrovní libovolný text, který však zpravidla bude pro názornost označovat číslo úrovně. Například následující tři řádky definic

```
--;L0; .Celkem *****; za případným středníkem jen komentář  
--;L1; .Závod =====&ZAVOD =====; ZAVOD  
--;L2; .Sklad -----&zavod --&sklad -----; SKLAD
```

určují dvě skupinové úrovně L2, L1 a koncovou úroveň L0. Hierarchie úrovní je dána pořadím zápisu definičních řádků, ne číslem u písmena L. Poslední řádek určuje nejnižší úroveň.

### Definice sumarizovaných sloupců

Aby bylo co sumarizovat (střádat), musí být zadána jména sloupců se svým jedním nebo až pěti sumarizačními typy. K tomu je ovšem také nutné zadat nějaké skupinové úrovně (L0, L1, ...).

Zadání sumarizace spočívá v zápisu jednoho nebo více definičních řádků **--;S**. Definiční řádky nemusí být nijak zvlášť seřazeny. Každý definiční řádek začíná čtyřznakovým



symbolem **--;S** v 1. pozici a pokračuje zápisy oddělenými středníkem. Zápisy určují jméno sloupce a způsob, jakým se budou jeho hodnoty střádat. Definiční řádek má tvar

```
--;S [střádaný-sloupec [; ind] [; ind] [; ind] [; ind] [; ind] [;]]
```

kde *ind* může být jeden ze symbolů – indikátorů S, A, M, m, C, které znamenají součet, průměr, maximum, minimum, počet. Indikátory nemusí být seřazeny v *žádném určitém pořadí*, nemusí být zadány všechny, a dokonce nemusí být zadány žádné. Sumární řádky ve výsledku dotazu se ovšem řadí v pořadí: S - součet, A - průměr, M - maximum, m - minimum, C - počet).

Je-li v některé pozici indikátor prázdný nebo chybný, má se za to, že není zadán.

Není-li v definičním řádku --;S zadán žádný indikátor nebo není-li definiční řádek --;S vůbec zadán, zpracování úrovní se neprovádí, i když je třeba zadáno definicemi --;L. Rovněž tak obráceně: Není-li zadán žádný definiční řádek --;L, zpracování úrovní se neprovádí, i když je třeba zadán definiční řádek --;S s nějakým indikátorem.

U sloupců typu DECIMAL, NUMERIC, INTEGER, SMALLINT a BIGINT lze zjišťovat součet, průměr, maximum, minimum a počet. U sloupců typu CHAR, VARCHAR, VARCHAR2, DATE, TIME a TIMESTAMP lze zjišťovat pouze maximum, minimum a počet.

*Hodnoty NULL* se do sumarizace nezahrnují.

**Poznámka:** Je-li chybně zadán součet (S) nebo průměr (A) pro *nenumernický* sloupec, pochopitelně se pro něj žádná hodnota netiskne. Řádek s úvodním textem (viz dále) se však vytiskne (i když třeba zároveň není sumarizován žádný *numerický* sloupec).

Například definice

```
--;S MNOZSTVI; C
--;S CELKEM ; S ; A ; M ; m
```

určují dva střádané sloupce, přičemž u sloupce MNOZSTVI se vyhodnocuje jen počet řádků (C) a u sloupce CELKEM se pořizuje součet (S), průměr (A), maximum (M) a minimum (m).

Úvodní texty pro sumární řádky

Standardní označení (zkratky) sumárních hodnot jsou SUM pro součet, AVG pro průměr (average), MAX pro maximum, MIN pro minimum a COUNT pro počet. Lze nahradit jinými pomocí definičního řádku **--;s**.

Definiční řádek začíná čtyřznakovým symbolem **--;s** (malé písmeno s) v 1. pozici. Má tvar

```
--;s [ [sum-text]; [avg-text]; [max-text]; [min-text]; [count-text] [;] ]
```

kde texty zapsané v daných pozicích nahradí standardní označení. Zde je ovšem pořadí zápisů podstatné. Jestliže některý z textů chybí, uplatní se v dané pozici standardní označení.

Například definiční řádek

```
--;s Součet ; Průměr ; Maximum ; Minimum ; Počet
```

nahradí všechny standardní zkratky.

## Příklad 1

V následujícím příkladu je ilustrována sumarizace *detailního* dotazu, tj. takového, kde není použito fráze GROUP BY. V titulních řádcích jsou zařazeny *proměnné* (dokonce jedna proměnná vícekrát). Dva sloupce budou *vynechány* z výstupu (ZAVOD, NAZ). Sumarizují se dva sloupce (MNOZSTVI, CELKEM) ve třech skupinových úrovních L0 (celkem), L1 (ZAVOD), L2 (SKLAD).

Text skriptu:

```
--zboží podle skladů a závodů

--;t >>> Závod 4x:      &ZAVOD , &ZAVOD , &ZAVOD , &ZAVOD .          <<<
--;t >>> Název zboží: &NAZ ,   Sklad: &SKLAD . <<<
--;t

--;HZav;Skl;Č.zboží;Název zboží;Cena/j.;Množství;   Cena celkem;
--;H===;===;=====;=====;=====;=====;
--;H

--;T 1; 1;   ; 3

--;O ZAVOD
--;O NAZ;

--;S MNOZSTVI; C
--;S CELKEM ; S ; A ; M ; m

--;s Součet ; Průměr ; Maximum ; Minimum ; Počet

--;L0;   .Celkem *****; za případným středníkem jen komentář
--;L1;   .Závod =====&ZAVOD =====; ZAVOD
--;L2;   .Sklad -----&zavod --&sklad -----; SKLAD

--;P a4; fs9; P; 10; 10; 10; 10

SELECT
  S.ZAVOD,
  S.SKLAD,
  S.CZBOZI,
  SUBSTR(C.NAZZBO, 1, 15) AS NAZ,
  DEC(C.CENAJ, 12, 2) AS CJ,
  S.MNOZSTVI,
  DECIMAL ( C.CENAJ*S.MNOZSTVI, 12, 5 ) AS CELKEM
FROM STAVY AS S
  JOIN CENY AS C ON S.CZBOZI = C.CZBOZI
  ORDER BY S.ZAVOD, S.SKLAD, S.CZBOZI
```

## Výsledek skriptu:

Zboží podle skladů a závodů

čtvrtek 11. prosince 2025 17:28:01

>>> Závod 4x: 01, 01, 01, 01. <<<  
>>> Název zboží: PIŠKOTY OPAVIA, Sklad: 01. <<<

| Sk1 | Č.zboží | Cena/j. | Množství  | Cena celkem |
|-----|---------|---------|-----------|-------------|
| === | =====   | =====   | =====     | =====       |
| 01  | 00001   | 8,99    | 1 579,444 | 14 199,202  |
| 01  | 00002   | 459     | 1         | 459         |
| 01  | 00010   | 6 500   | 7         | 45 500      |

.Sklad -----01--01-----

|         |            |
|---------|------------|
| Součet  | 60 158,202 |
| Průměr  | 20 052,734 |
| Maximum | 45 500     |
| Minimum | 459        |
| Počet   | 3          |

|    |       |       |      |        |
|----|-------|-------|------|--------|
| 02 | 00003 | 1,25  | 19   | 23,75  |
| 02 | 00009 | 250   | 1,08 | 270    |
| 02 | 00010 | 6 500 | 3    | 19 500 |

.Sklad -----01--02-----

|         |           |
|---------|-----------|
| Součet  | 19 793,75 |
| Průměr  | 6 597,917 |
| Maximum | 19 500    |
| Minimum | 23,75     |
| Počet   | 3         |

|    |       |      |   |      |
|----|-------|------|---|------|
| 03 | 00003 | 1,25 | 1 | 1,25 |
|----|-------|------|---|------|

.Sklad -----01--03-----

|         |      |
|---------|------|
| Součet  | 1,25 |
| Průměr  | 1,25 |
| Maximum | 1,25 |
| Minimum | 1,25 |
| Počet   | 1    |

.Závod =====01=====

|         |            |
|---------|------------|
| Součet  | 79 953,202 |
| Průměr  | 11 421,886 |
| Maximum | 45 500     |
| Minimum | 1,25       |
| Počet   | 7          |

|    |       |       |   |       |
|----|-------|-------|---|-------|
| 01 | 00005 | 120   | 2 | 240   |
| 01 | 00006 | 10,55 | 2 | 21,1  |
| 01 | 00008 | 1 700 | 2 | 3 400 |

.Sklad -----02--01-----

|         |           |
|---------|-----------|
| Součet  | 3 661,1   |
| Průměr  | 1 220,367 |
| Maximum | 3 400     |
| Minimum | 21,1      |
| Počet   | 3         |

|    |       |       |   |       |
|----|-------|-------|---|-------|
| 02 | 00009 | 250   | 2 | 500   |
| 02 | 00011 | 159   | 2 | 318   |
| 02 | 00014 | 3 500 | 2 | 7 000 |
| 02 | 00018 | 56    | 2 | 112   |

.Sklad -----02--02-----

|        |       |
|--------|-------|
| Součet | 7 930 |
|--------|-------|

</

|         |   |       |
|---------|---|-------|
| Maximum |   | 7 000 |
| Minimum |   | 112   |
| Počet   | 4 |       |

.Závod =====02=====

|         |   |           |
|---------|---|-----------|
| Součet  |   | 11 591,1  |
| Průměr  |   | 1 655,871 |
| Maximum |   | 7 000     |
| Minimum |   | 21,1      |
| Počet   | 7 |           |

.Celkem \*\*\*\*\*

|         |    |            |
|---------|----|------------|
| Součet  |    | 91 544,302 |
| Průměr  |    | 6 538,879  |
| Maximum |    | 45 500     |
| Minimum |    | 1,25       |
| Počet   | 14 |            |

## Příklad 2

Ve skriptu SUMARIZACE je nejdůležitější kód NP zapsaný u sloupce ZAVOD v definici L1. Způsobuje *skok na novou stránku*, jestliže se změní skupina jednoho závodu na skupinu jiného závodu. Kód NP lze zapsat ke každé úrovni kromě L0.

Text skriptu:

```
--Sumarizace zboží podle skladů a závodů, závod na nové stránce
```

| Zav               | Skl | Č. zboží | Název zboží     | Cena/j. | Množství | Cena celkem |
|-------------------|-----|----------|-----------------|---------|----------|-------------|
| 02                | 01  | 00005    | Tričko bílé     | 120,00  | 2,00     | 240,00      |
|                   |     | 00006    | Ponožky pánské  | 10,55   | 2,00     | 21,10       |
|                   |     | 00008    | Kalhoty džínové | 1700,00 | 2,00     | 3400,00     |
| .Sklad 02 01      |     |          |                 |         |          |             |
|                   |     |          |                 |         | Součet   | 3661,10     |
|                   |     |          |                 |         | Počet    | 3           |
|                   | 02  | 00009    | Whisky Balantin | 250,00  | 2,00     | 500,00      |
|                   |     | 00011    | Taška sportovní | 159,00  | 2,00     | 318,00      |
|                   |     | 00014    | Sako tvidové, n | 3500,00 | 2,00     | 7000,00     |
|                   |     | 00018    | Husí sádlo v ko | 56,00   | 2,00     | 112,00      |
| .Sklad 02 02      |     |          |                 |         |          |             |
|                   |     |          |                 |         | Součet   | 7930,00     |
|                   |     |          |                 |         | Počet    | 4           |
| .Závod 02         |     |          |                 |         |          |             |
|                   |     |          |                 |         | Součet   | 11591,10    |
|                   |     |          |                 |         | Počet    | 7           |
| .CELKEM ZA PODNIK |     |          |                 |         |          |             |
|                   |     |          |                 |         | Součet   | 25311,59    |
|                   |     |          |                 |         | Počet    | 14          |

Strana 2

```
--;HZav;Skl;Č.zboží;Název zboží;Cena/j.;Množství; Cena celkem;
--;H===;===;=====;=====;=====;=====;
--;H
```

```
--;T 1; 1; - ; 2; ZAVOD; SKLAD
```

```

--;S CELKEM ; S ; C

--;s Součet ; Průměr ; Maximum ; Minimum ; Počet

--;L0; .CELKEM ZA PODNIK
--;L1; .Závod &ZAVOD ; ZAVOD; NP
--;L2; .Sklad &ZAVOD &SKLAD ; SKLAD

--;P a4; fs9; L; lm5; rm5; tm5; bm5

SELECT
    S.ZAVOD -- Číslo závodu
    ,S.SKLAD -- Číslo skladu
    ,S.CZBOZI -- Číslo zboží
    ,SUBSTR(C.NAZZBO, 1, 30) AS NAZ -- Název zboží
    ,DEC(C.CENAJ, 12, 2) AS CJ -- Cena za jednotku
    ,S.MNOZSTVI -- Množství jednotek
    ,DECIMAL ( C.CENAJ*S.MNOZSTVI, 12, 5 ) AS CELKEM -- Cena množství

FROM STAVY AS S
    JOIN CENY4 AS C ON S.CZBOZI = C.CZBOZI
ORDER BY S.ZAVOD, S.SKLAD, C.CZBOZI

```

Výsledek skriptu v PDF:

Podoba na tiskárně (simulace v macOS):

Java Printing

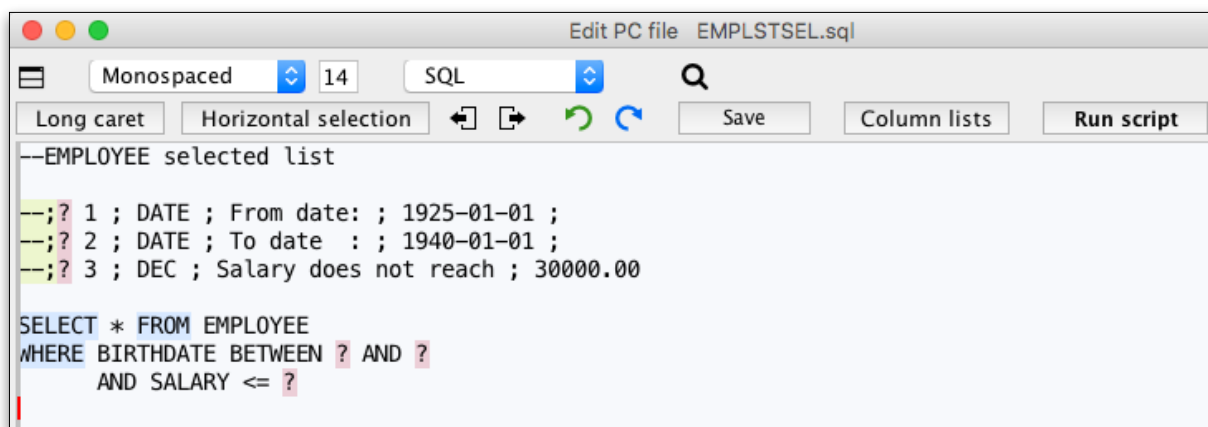
Sumarizace zboží (CENY) podle skladů a závodů, závod na nové stránce

Úterý, 15. prosince 2015 18:05:02

| <u>Zav</u> | <u>Skl</u> | <u>Č.zboží</u> | <u>Název zboží</u> | <u>Cena/j.</u> | <u>Množství</u> | <u>Cena celkem</u> |
|------------|------------|----------------|--------------------|----------------|-----------------|--------------------|
| 01         | 01         | 00001          | PIŠKOTY OPAVIA     | 8,99           | 1,00            | 8,99               |
|            |            | 00002          | Zubní pasta Kal    | 459,00         | 1,00            | 459,00             |
|            |            | 00010          | Koňak Gruzínský    | 6500,00        | 1,00            | 6500,00            |
| .Sklad     |            | 01 01          |                    |                | Součet<br>Počet | 6967,99<br>3       |
|            | 02         | 00003          | Prádelní šňůra     | 1,25           | 1,00            | 1,25               |
|            |            | 00009          | Whisky Balantin    | 250,00         | 1,00            | 250,00             |
|            |            | 00010          | Koňak Gruzínský    | 6500,00        | 1,00            | 6500,00            |
| .Sklad     |            | 01 02          |                    |                | Součet<br>Počet | 6751,25<br>3       |
|            | 03         | 00003          | Prádelní šňůra     | 1,25           | 1,00            | 1,25               |
| .Sklad     |            | 01 03          |                    |                | Součet<br>Počet | 1,25<br>1          |
| .Závod     |            | 01             |                    |                | Součet<br>Počet | 13720,49<br>7      |

Strana 1

## Editace skriptů




- *Dělení/spojení* - střídaté tlačítko rozdělí text editoru na horní a spodní oblast vodorovnou čarou nebo obě oblasti spojí. Ve spodní oblasti je kopie textu editoru. Všechny změny provedené v jedné oblasti se automaticky promítají do druhé oblasti.
- *Lucida Console aj.* – z rozbalovacího seznamu vyberte písmo pro text v editoru.
- *SQL nebo \*NONE.* – z rozbalovacího seznamu vyberte SQL ke zvýraznění bloků (složených příkazů).
- *Find text* – vyvolá okno k vyhledání textu. Lze také použít klávesovou zkratku Ctrl F.
- *Long caret/Short caret* – určí ukazatel místa v textu jako dlouhou svislou přímku nebo standardní krátkou svislou čárku.
- *Horizontal/Vertical selection* – určuje způsob výběru textu. *Horizontal selection* je běžný způsob v PC editorech. *Vertical selection* vybírá v textu obdélníkovou oblast.
- *Posuv výběru* – tlačítko posune vybraný text o jednu pozici vlevo, tlačítko posune vybraný text o jednu pozici vpravo. Lze také použít klávesové zkratky Ctrl ← a Ctrl →.
- *Undo* – odvolat - odstranit změny. Lze také použít klávesovou zkratku Ctrl Z.
- *Redo* – přivolat - obnovit změny. Lze také použít klávesovou zkratku Ctrl Y.
- *Save* – uloží změny. Lze také použít klávesovou zkratku Ctrl S.
- *Column lists* – vyvolá okno, kde po výběru konkrétního schématu a tabulky (pohledu) se ukáže seznam sloupců. Seznam sloupců lze modifikovat a editovat (přidáním čárek) a kopírovat do oblasti editoru.
- *Run script* – spouští SQL script.

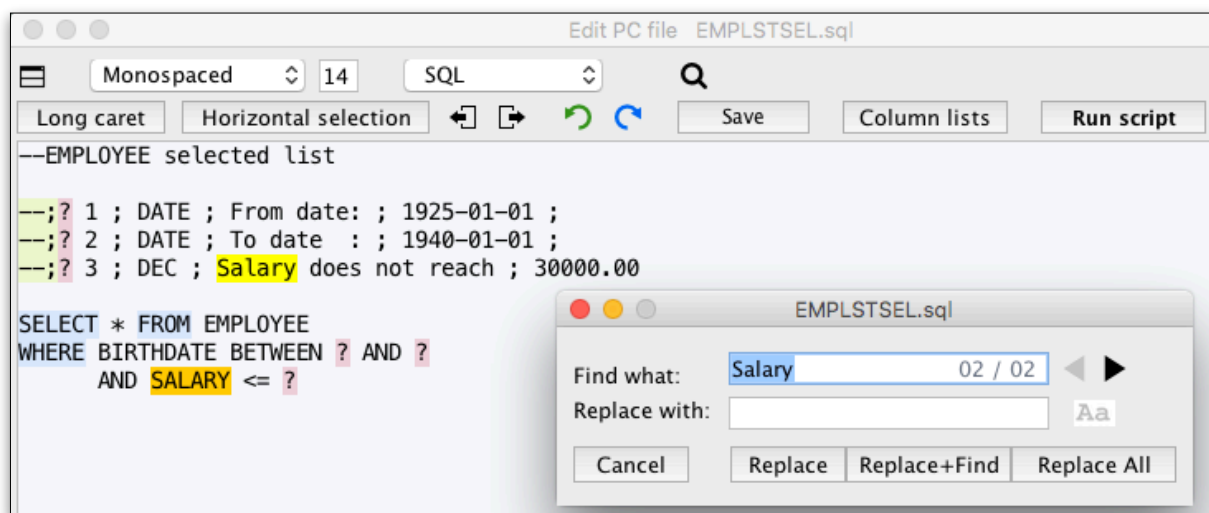
*Klávesa ESC* ukončí editaci (bez uložení) a odstraní okno.



Poznámka: V systému macOS se použije klávesa *Cmd* místo Ctrl.



## Hledání textu



Klepnutí na lupu  nebo stisk klávesové kombinace Ctrl F (Cmd F v systému macOS) vyvolá okno k vyhledání textu.



- *Find what* – запиšte textový vzorek k vyhledání. Čísla v poli ukazují pořadové číslo a počet shod.
-   *Tlačítka s šipkami* – vyhledávají předchozí a následující shodný vzorek v souboru. Po klepnutí šipka zčerná a ukazuje směr hledání a nahrazování. Opačná šipka zešedne. Stejnou funkci mají klávesové zkratky Ctrl ↑ (předchozí) a Ctrl ↓ (další).
- *Replace with* – náhradní text.
- **Aa/Aa** *střídavé tlačítko* – je-li světle šedé, nepožaduje shodu velkých a malých písmen, je-li černé, musí se vyhledaný text přesně shodovat se vzorkem.
- *Replace* – nahradí právě nalezený text náhradním textem.
- *Replace+Find* – nahradí právě nalezený text náhradním textem a nalezne *následující* shodný text.
- *Replace All* – nahradí *všechny* shodné texty náhradním textem.

Poznámka: V systému macOS se použije klávesa *Cmd* místo Ctrl.

## Shifting selected text

Tlačítka  a  posunou vybraný text vlevo nebo vpravo o jednu pozici. Stejnou funkci dělají klávesové zkratky Ctrl ← a Ctrl →. Výsledky posunu mohou být odvolány nebo přivolány.

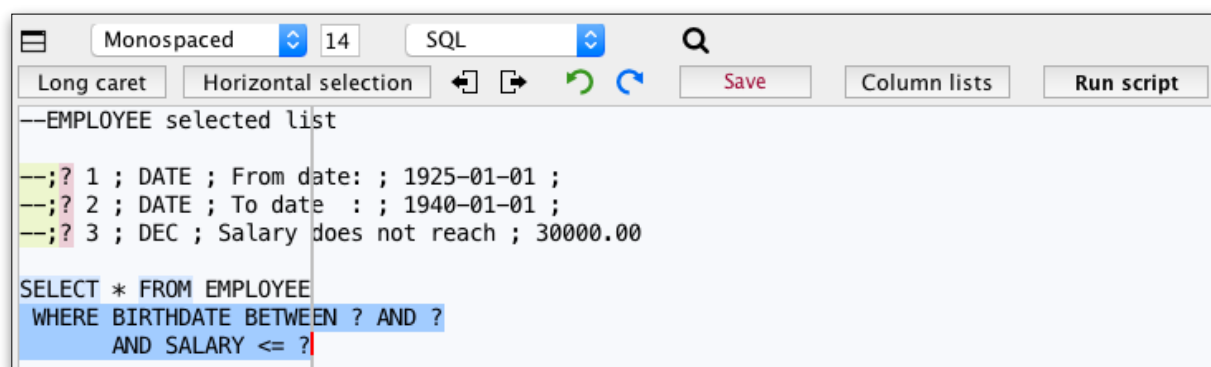
Poznámka: V systému macOS se použije klávesa *Cmd* místo Ctrl.

## Horizontal selection

Vybraný text se posune o jednu pozici *vpravo*.

Vybraný text se posune o jednu pozici *vlevo* zároveň se zbytkem řádků, jestliže začíná na začátku řádku a jestliže všechny řádky vybraného textu obsahují alespoň jednu mezeru na začátku.

V následujícím obrázku byl vybraný text posunut vpravo.

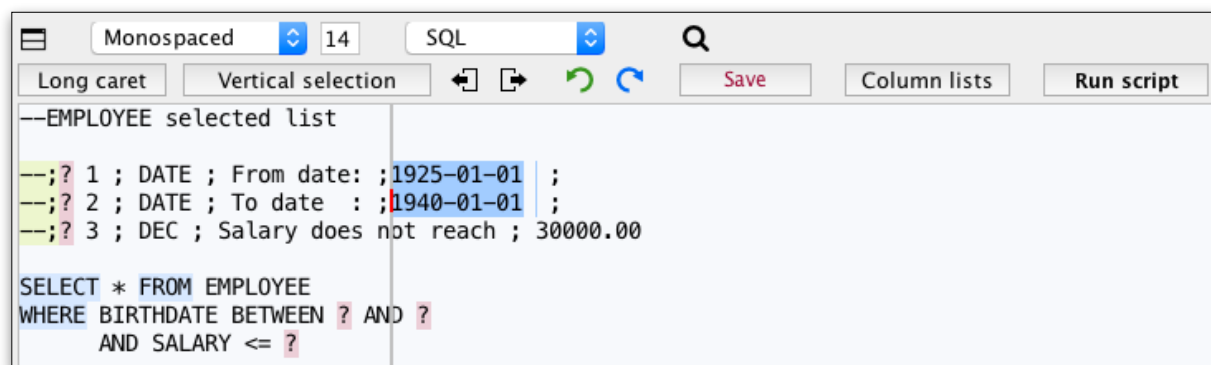


## Vertical selection

Vybraný *obdélník* se posune *vlevo*, jestliže v nevybrané části zbývá alespoň jeden sloupec mezer vlevo od obdélníku. Obdélník přepíše sloupec vlevo a nechá za sebou (vpravo) sloupec mezer.

Vybraný *obdélník* se posune *vpravo* a nechá za sebou sloupec mezer (vlevo).

V následujícím obrázku byl vybraný obdélník posunut vlevo.



## Copy, cut and paste selected text

Ke kopírování, vyjímání a vkládání se používají běžné klávesové zkratky Ctrl C, Ctrl X a Ctrl V.

Operace Copy a Cut kopírují vybraný text do *schránky operačního systému*.

Operace Paste přečte data ze schránky a vloží je na požadované místo. To může být v oblasti editoru, anebo někde jinde v PC.

Výsledky těchto operací mohou být odvolány a přivolány.

Poznámka: V systému macOS se použije klávesa *Cmd* místo Ctrl.

## Horizontal selection

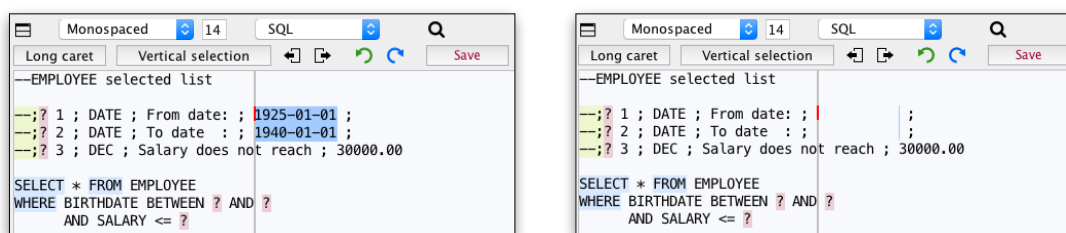
Všechny tyto operace fungují jako v PC.

## Vertical selection

Operace **Copy** (kopírovat) zkopíruje výběr do interní oblasti a také do systémové schránky.

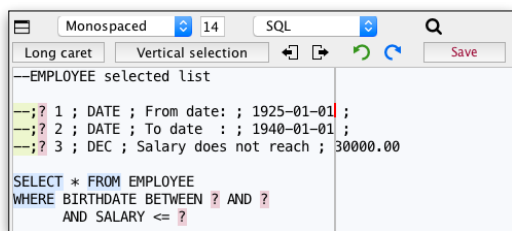
Operace **Cut** (vyjmout) zkopíruje a uloží výběr do interní oblasti a také do systémové schránky, pak *obdélníkovou oblast vyčistí* (vloží tam mezery).

Například obdélník z levého obrázku byl vyňat operací Cut:

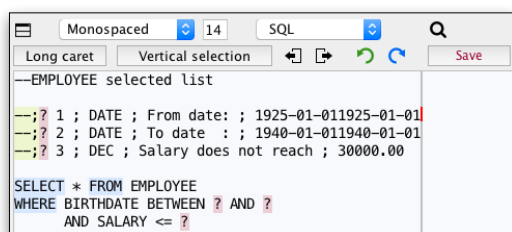


Všimněme si pozice *ukazatele*, stojí na *začátku* vyňatého obdélníku. Modrá čára vpravo označuje pravý okraj obdélníku. Chceme-li, můžeme následnou operací *Paste* vložit vymazaná data zpět do jejich původních pozic.

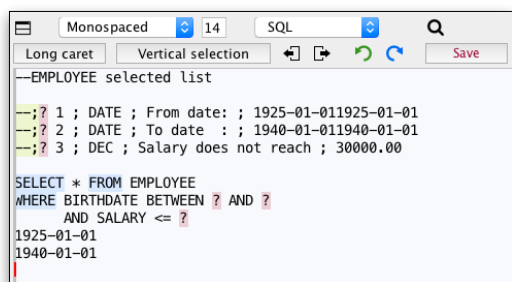
Operace **Paste** *přepíše* oblast v editoru *vpravo a dolů* od pozice zvolené ukazatelem. Může to být oblast původně kopírovaná nebo vyňatá (operací Copy nebo Cut):



Tentokrát stojí *ukazatel* v *pravém horním rohu* obdélníku. Bezprostředně následující operace *Paste* vloží data napravo od ukazatele:



Je-li oblast editoru kratší než vkládaný obdélník, doplní se potřebné prázdné řádky a vkládaná data se do nich vloží:

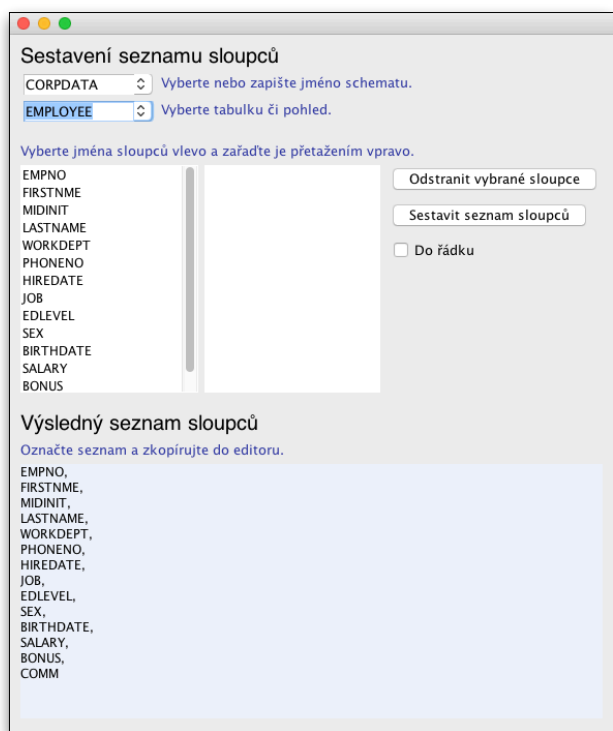


## Column lists – Seznamy sloupců

Tato funkce usnadňuje editaci skriptů tím, že získává seznamy sloupců z objektů IBM i.

Tlačítko Column lists vyvolá okno, kde uživatel vybere schema a tabulku (v obrázcích CORPDATA a EMPLOYEE).

Tato akce vyplní levý rámeček seznamem jmen sloupců vzatých z tabulky (EMPLOYEE).



Spodní část okna – "Resulting column list" – nyní obsahuje jména sloupců s čárkami mezi nimi a každé jméno na zvláštním řádku. Tento seznam, buď celý nebo jeho část, lze vybrat a přímo *zkopírovat* nebo *přetáhnout* do oblasti editoru. K tomu může sloužit i kontextová nabídka v levém nebo pravém rámečku.

## Odstranění vybraných sloupců

Tlačítko *Odstranit vybrané sloupce* lze použít ke změně seznamu v pravém rámečku odstraněním vybraných jmen a nejspíš přetažením jiných jmen z levého rámečku.

## Sestavení seznamu sloupců

**Sestavení seznamu sloupců**

CORPDATA ▾ Vyberte nebo zapište jméno schematu.

EMPLOYEE ▾ Vyberte tabulku či pohled.

Vyberte jména sloupců vlevo a zařaďte je přetažením vpravo.

|                                                                                                                                       |                                         |
|---------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------|
| EMPNO<br>FIRSTNAME<br>MIDINIT<br>LASTNAME<br>WORKDEPT<br>PHONENO<br>HIREDATE<br>JOB<br>EDLEVEL<br>SEX<br>BIRTHDATE<br>SALARY<br>BONUS | WORKDEPT<br>EMPNO<br>LASTNAME<br>SALARY |
|---------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------|

Odstranit vybrané sloupce

Sestavit seznam sloupců

☐ Do řádku

**Výsledný seznam sloupců**

Označte seznam a zkopírujte do editoru.

WORKDEPT,  
EMPNO,  
LASTNAME,  
SALARY

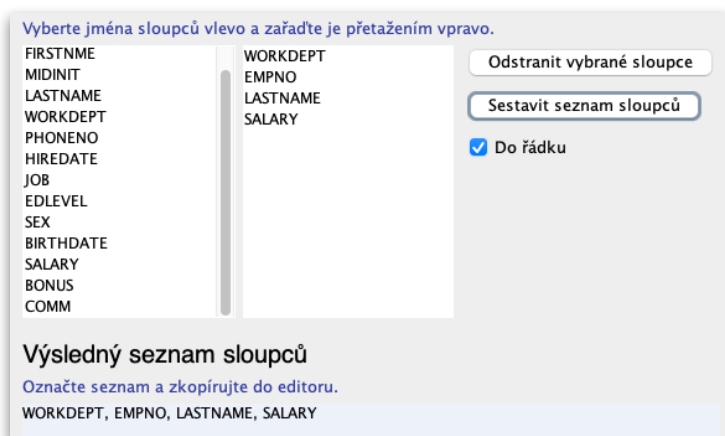
Jednotlivá jména nebo skupiny jmen lze *přetáhnout* z levého do *pravého rámečku*.

Tlačítko *Sestavit seznam sloupců* vyprodukuje ze seznamu v *pravém rámečku* editovaný seznam sloupců v části "Výsledný seznam sloupců". Seznam je uspořádán do samostatných řádků, jestliže zaškrtačací políčko *Do řádku* není zaškrtnuto.

Tento seznam, buď celý nebo jeho část, lze vybrat a přímo *zkopírovat* nebo *přetáhnout* do oblasti editoru.

## Sestavení seznamu sloupců do řádku

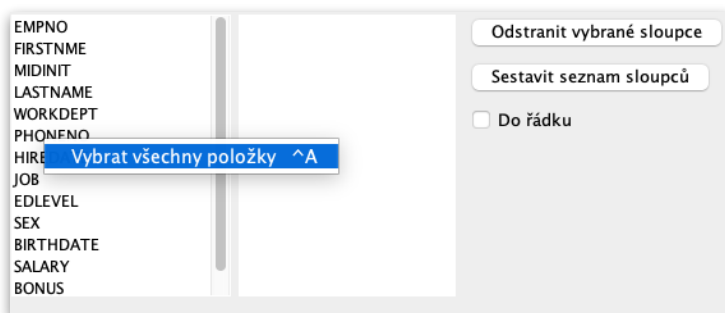
Je-li zaškrtnuto políčko *Do řádku* sestaví tlačítko *Sestavit seznam sloupců* editovaný seznam do jednoho řádku.



Tento seznam, buď celý nebo jeho část, lze vybrat a přímo *zkopírovat* nebo *přetáhnout* do oblasti editoru.

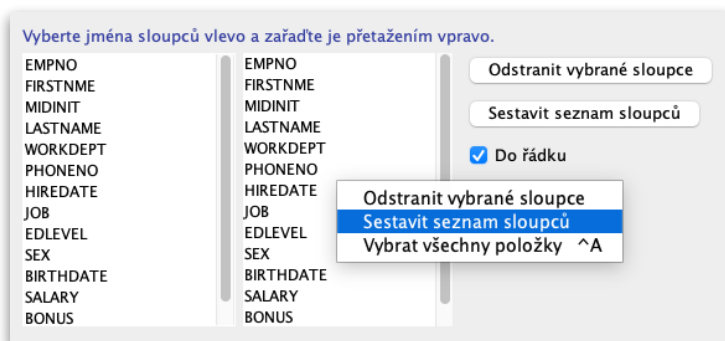
### Výběr všech jmen v levém i pravém rámečku

- Klepnutím pravým tlačítkem myši a vybrat příkaz *Vybrat všechny položky* v kontextové nabídce.
- Stiskem klávesové zkratky Ctrl A (Cmd A in macOS).



### Kontextová nabídka na pravém rámečku

- *Odstranit vybrané sloupce*. Účinkuje stejně jako stejně nazvané tlačítko.
- *Sestavit seznam sloupců*. Účinkuje stejně jako stejně nazvané tlačítko.
- *Vybrat všechny položky*. Účinkuje stejně jako klávesová zkratka Ctrl A (Cmd A v macOS).



## Práce s členy fyzických a logických souborů

Na příkladu ukážeme, jak lze pracovat s datovými členy (members) fyzických a logických souborů. V případě logického souboru lze pracovat se všemi logickými členy spojenými do jednoho.

Předpokladem je, že soubory jsou vytvořeny pomocí DDS (Data Description Specifications). Vytvoření tabulek pomocí příkazů CREATE TABLE nebo CREATE VIEW totiž neumožňuje dodatečně přidat nebo odstranit datové členy.

### Vytvoření fyzického a logického souboru se členy

Vytvořím fyzický soubor CENY.

```
*****
*   Soubor CENY - Ceník zboží
*****
A                                     UNIQUE
A           R CENYR
*   Číslo zboží
A           CZBOZI           5A
*   Cena za jednotku (kus)
A           CENAJ           9P 2
*   Název zboží
A           NAZZBO           50A           CCSID(1208)
*   Definice klíče - Číslo zboží
A           K CZBOZI
```

Vytvořím logický soubor CENYL.

```
*****
*   Soubor CENY
*   Logický soubor
*****
A           R CENYR           PFILE(CENY)
A                                     FORMAT(CENY)
A           K CZBOZI
```

V souboru CENY definuji maximální počet členů.

```
CHGPF      FILE(CENY)  MAXMBRS(12)
```

Přidám tři členy k souboru CENY.

```
ADDPFM      FILE(VZTOOL/CENY)  MBR(CENY_01)
ADDPFM      FILE(VZTOOL/CENY)  MBR(CENY_02)
ADDPFM      FILE(VZTOOL/CENY)  MBR(CENY_03)
```

V logickém souboru CENYL definuji maximální počet členů.

```
CHGLF      FILE(VZTOOL/CENYL)  MAXMBRS(12)
```

K logickému souboru CENYL přidám také tři členy.

```
ADDLFM      FILE(VZTOOL/CENYL)  MBR(CENYL_01)  DTAMBR((VZTOOL/CENY (CENY_01)))
ADDLFM      FILE(VZTOOL/CENYL)  MBR(CENYL_02)  DTAMBR((VZTOOL/CENY (CENY_02)))
ADDLFM      FILE(VZTOOL/CENYL)  MBR(CENYL_03)  DTAMBR((VZTOOL/CENY (CENY_03)))
```

## Skript k vytvoření alias objektů a zápisu dat do členů

Následující skript CENY\_ALIASES smaže alias objekty (jestliže existují) a vytvoří nové, pro každý člen fyzického i logického souboru pojmenované jako členy.

```
/* Předem odstraním objekty ALIAS pro členy fyzického a logického souboru */
DROP ALIAS VZTOOL.CENY_01 ;
DROP ALIAS VZTOOL.CENY_02 ;
DROP ALIAS VZTOOL.CENY_03 ;
DROP ALIAS VZTOOL.CENYL_01 ;
DROP ALIAS VZTOOL.CENYL_02 ;
DROP ALIAS VZTOOL.CENYL_03 ;

/* Vytvořím objekty ALIAS pro členy souboru CENY */
CREATE ALIAS VZTOOL.CENY_01 FOR VZTOOL.CENY(CENY_01) ;
CREATE ALIAS VZTOOL.CENY_02 FOR VZTOOL.CENY(CENY_02) ;
CREATE ALIAS VZTOOL.CENY_03 FOR VZTOOL.CENY(CENY_03) ;

/* Vytvořím objekty ALIAS pro členy logického souboru CENYL */
CREATE ALIAS VZTOOL.CENYL_01 FOR VZTOOL.CENY(CENY_01) ;
CREATE ALIAS VZTOOL.CENYL_02 FOR VZTOOL.CENY(CENY_02) ;
CREATE ALIAS VZTOOL.CENYL_03 FOR VZTOOL.CENY(CENY_03) ;

/* Vrátím předvolené schema */
SET SCHEMA DEFAULT;

/* Vymažu záznamy ze členů souboru CENY */
DELETE FROM CENY_01 ;
DELETE FROM CENY_02 ;
DELETE FROM CENY_03 ;

/* Vložím záznamy do členů souboru CENY */
INSERT INTO CENY_01 values ('00001', 8,99, 'PIŠKOTY OPAVIA') ;
INSERT INTO CENY_01 values ('00002', 459,00, 'Zubní pasta Kalodont') ;
INSERT INTO CENY_01 values ('00003', 1,25, 'Prádelní šňůra') ;

INSERT INTO CENY_02 values ('00004', 10,50, 'Ponožky pánské tmavé') ;
INSERT INTO CENY_02 values ('00005', 120,00, 'Tričko bílé') ;
INSERT INTO CENY_02 values ('00006', 10,55, 'Ponožky pánské bílé, nové')
```

Provedením skriptu vzniknou ALIAS objekty. ALIAS objekty mají v systému typ \*FILE a atribut DDMF. DDMF (Distributed Data Management File) je objekt, který slouží k přístupu ke vzdálenému místu (Remote Location) z lokálního místa (Local Location). Jde o objekt z architektury SNA, APPC. Zde jsou obě místa totožná a objekt slouží jako prostředek přístupu k datovému souboru.

| Opt | Object   | Type  | Attribute | Text  |
|-----|----------|-------|-----------|-------|
| —   | CENY     | *FILE | PF-DTA    | CENY  |
| —   | CENY_01  | *FILE | DDMF      |       |
| —   | CENY_02  | *FILE | DDMF      |       |
| —   | CENY_03  | *FILE | DDMF      |       |
| —   | CENYL    | *FILE | LF        | CENYL |
| —   | CENYL_01 | *FILE | DDMF      |       |
| —   | CENYL_02 | *FILE | DDMF      |       |
| —   | CENYL_03 | *FILE | DDMF      |       |



## Kontrolní výpis dat ze členů logického souboru

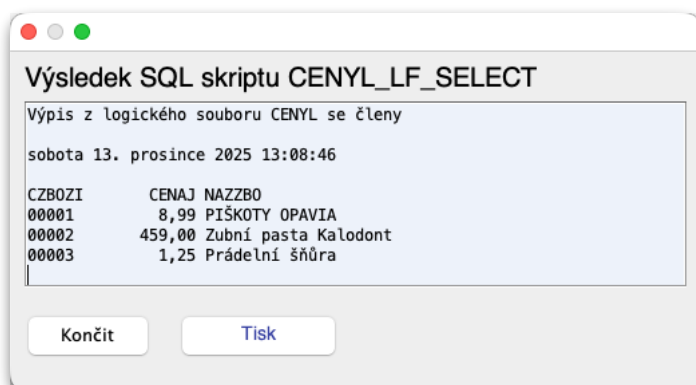
Nyní provedeme dotazy na logické členy.

--Výpis z logického souboru CENYL se členy

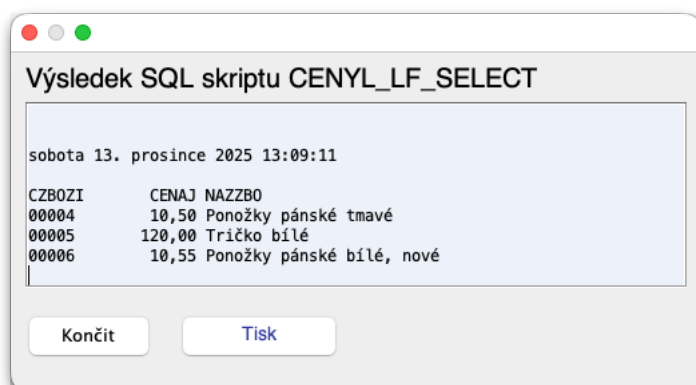
```
SELECT * FROM CENYL_01;  
SELECT * FROM CENYL_02;  
SELECT * FROM CENYL_03;  
SELECT * FROM CENYL
```

Dostaneme čtyři obrazovky s obsahem logických členů.

Člen CENYL\_01:



Člen CENYL\_02:



Člen CENYL\_03 (prázdný):



Všechny členy dohromady:

