

IBM i Programming Tool

Návod k použití

Obsah

Obsah.....	2
Úvod.....	4
Obsah aplikačního adresáře.....	4
Spuštění aplikace	5
Přehled.....	5
Editace souborů.....	6
<i>Hledání textu</i>	<i>7</i>
<i>Zvýraznění bloků</i>	<i>8</i>
<i>Posouvání vybraného textu.....</i>	<i>10</i>
Horizontal selection.....	10
Vertical selection	11
<i>Copy, cut and paste selected text</i>	<i>12</i>
Horizontal selection.....	12
Vertical selection	12
<i>Pomůcka pro jazyky založené na formulářích.....</i>	<i>13</i>
Kompilace.....	14
<i>Source type</i>	<i>16</i>
Zdrojové členy	16
IFS soubory	16
<i>Compile command</i>	<i>17</i>
Uchování atributů komplikace	17
<i>Change library list.....</i>	<i>18</i>
Library pattern	18
Current library	18
Vytváření uživatelského seznamu knihoven	18
<i>Compiled object.....</i>	<i>19</i>
Library	19
Object	19
Library pattern	19
<i>Perform command</i>	<i>19</i>
<i>Spooled files.....</i>	<i>20</i>
<i>Job log.....</i>	<i>20</i>
Příklad komplikace	21
Parametry aplikace	24
<i>IBM i server a uživatel</i>	<i>24</i>
<i>Servers.....</i>	<i>25</i>
<i>Connect/Reconnect.....</i>	<i>27</i>
<i>Vstupní pole LIB, FILE, MBR</i>	<i>28</i>
<i>IBM i source type.....</i>	<i>28</i>
<i>PC charset.....</i>	<i>29</i>
<i>IBM i CCSID</i>	<i>29</i>
<i>Source line length.....</i>	<i>29</i>
<i>Complete source record</i>	<i>30</i>
<i>Overwrite data</i>	<i>30</i>
<i>Windows disks.....</i>	<i>30</i>
Souborové systémy jako stromy	31
<i>Rozevírání uzlů</i>	<i>31</i>
<i>Levý strom – PC.....</i>	<i>31</i>
<i>Pravý strom - IBM i.....</i>	<i>32</i>
Výběr knihovních objektů	33
Kontextové nabídky pro PC.....	34
Kontextové nabídky pro IBM i	35

Vytváření adresářů a souborů v IBM i.....	36
<i>Zdrojový fyzický soubor.....</i>	<i>36</i>
<i>Zdrojový člen.....</i>	<i>36</i>
<i>IFS adresář.....</i>	<i>36</i>
<i>IFS soubor.....</i>	<i>36</i>
Způsoby kopírování.....	37
Kopírování z IBM i do PC	37
<i>Zdrojový člen → PC soubor.....</i>	<i>37</i>
<i>Zdrojové členy → PC adresář přetažením</i>	<i>38</i>
<i>Zdrojový soubor→ PC adresář copy-paste</i>	<i>38</i>
<i>IFS soubor (stream file) → PC soubor</i>	<i>38</i>
<i>Ukládací soubor (save file) → PC soubor</i>	<i>38</i>
Kopírování z PC do IBM i	39
<i>PC soubor → zdrojový člen.....</i>	<i>39</i>
<i>PC soubor → IFS soubor</i>	<i>39</i>
<i>PC soubor → ukládací soubor (save file)</i>	<i>39</i>
Kopírování z IBM i do IBM i.....	40
<i>Zdrojový člen → zdrojový člen.....</i>	<i>40</i>
<i>Zdrojový člen → IFS soubor</i>	<i>40</i>
<i>IFS soubor → zdrojový člen</i>	<i>40</i>
<i>IFS soubor → IFS soubor.....</i>	<i>40</i>
<i>Ukládací soubor v knihovně → IFS</i>	<i>41</i>
<i>Ukládací soubor v IFS adresáři → knihovna</i>	<i>41</i>
<i>Knihovna → knihovna.....</i>	<i>41</i>
Kopírování z PC do PC.....	41
Změna kódování v IFS.....	41
Zobrazování textových souborů	43
Hledání ve více souborech	44
Zobrazování ostatních (netextových souborů	45
Přejmenování objektů	45
Zobrazování a editace v PC – znakové sady.....	46
Zobrazování a editace v IBM i – znakové sady	46
Tiskové soubory (spooled files).....	47

Úvod

Tato aplikace je určena programátorům v systému IBM i. Umožňuje práci s textovými daty, zejména programy a doprovodnými zdrojovými texty pro jazyky RPG, CL, COBOL, C, C++, SQL a také DDS, TBL, PNLGRP. Uživatel může

- zobrazovat a editovat soubory
- kompilovat programy
- vytvářet, přejmenovat, nebo mazat adresáře a soubory
- kopírovat soubory a adresáře

Programy aplikace jsou napsány v jazyku Java a vyžadují verzi *Java SE 8* nebo vyšší nainstalovanou v PC. Spolupracují s třídami soustavy *IBM Toolbox for Java* (nebo JTOpen). Tyto třídy vyžadují, aby v IBM i běžely "host servery" a aby byl aktivován profil QUSER.

Aplikace byla vytvořena a testována v systému macOS a v systému Windows (v emulaci Parallels Desktop). Přitom bylo použito vzdálené internetové připojení k systému IBM i, verze 7.5.

Obsah aplikačního adresáře

Aplikace se neinstaluje. Je tvořena adresářem obsahujícím další adresáře a soubory, z nichž jsou pro uživatele užitečné tyto:

- Hlavní, spouštěcí soubor je ***IBMiProgTool.jar***. Otevřením tohoto souboru nebo poklepáním primárního tlačítka myši na tento soubor se aplikace spustí.
- Adresář *helpfiles* obsahuje soubory pro návod - help (tentotext) a formuláře pro jazyky RPG III, RPG IV, COBOL, DDS a také číslování sloupců.
- Adresář *icons* obsahuje ikony a obrázky pro použití v oknech aplikace.
- Adresář *logfiles* - obsahuje textové soubory *err.txt* a *out.txt*, přesměrovaný výstup ze souborů System.err a System.out.
- Adresář *paramfiles* obsahuje soubor *Parameters.txt* s parametry aplikace.
- Adresář *workfiles* obsahuje pomocné soubory pro uchování parametrů od ukončení do nového běhu aplikace.
- Soubor *README.md* obsahuje úvodní popis aplikace a historii verzí.

Poznámka 1: Soubory *err.txt* a *out.txt* slouží ke zjištění původu případné chyby v programu. Zprávy se nevypisují na konzolu, ale to těchto souborů

Poznámka 2: V adresáři získaném stažením z internetu je obsaženo více souborů a adresářů. Ty mohou být bez škody smazány.

Spuštění aplikace

Aplikace se spouští poklepáním na soubor ***IBMiProgTools.jar***.

Poznámka: Nabídka ***Help*** je umístěna v řádku nabídek v aplikačním okně (Windows) nebo v řádku nabídek na horním okraji pracovní plochy (macOS).

Přehled

Tato aplikace umožňuje

- zobrazovat a editovat textové soubory v PC a IBM i,
- hledat text ve více souborech,
- kompilovat zdrojové členy (source members) a IFS soubory (stream files),
- zobrazovat tiskové soubory (spoiled files),
- vytvářet soubory a adresáře v PC a IBM i,
- odstraňovat soubory a adresáře v PC a IBM i,
- přejmenovávat soubory a adresáře v PC a IBM i,
- kopírovat soubory a adresáře mezi PC a IBM i,
- kopírovat, vyprazdňovat a odstraňovat knihovny.

Aplikace pracuje s objekty těchto typů v systému IBM i:

- | | |
|------------------------|--|
| • Source physical file | zdrojový soubor - objekt typu *FILE s atributem PF, |
| • Source member | člen zdrojového souboru, |
| • IFS directory | adresář v IFS (Integrated File System) - objekt typu *DIR, |
| • IFS stream file | proudový soubor - objekt typu *STMF, |
| • Output queue | tisková fronta - objekt typu *OUTQ a tiskové soubory v ní, |
| • Save file | ukládací soubor - objekt typu *FILE s atributem SAVF, |
| • Library | knihovna - objekt typu *LIB. |

Zdrojový soubor se v aplikaci chová podobně jako *adresář*.

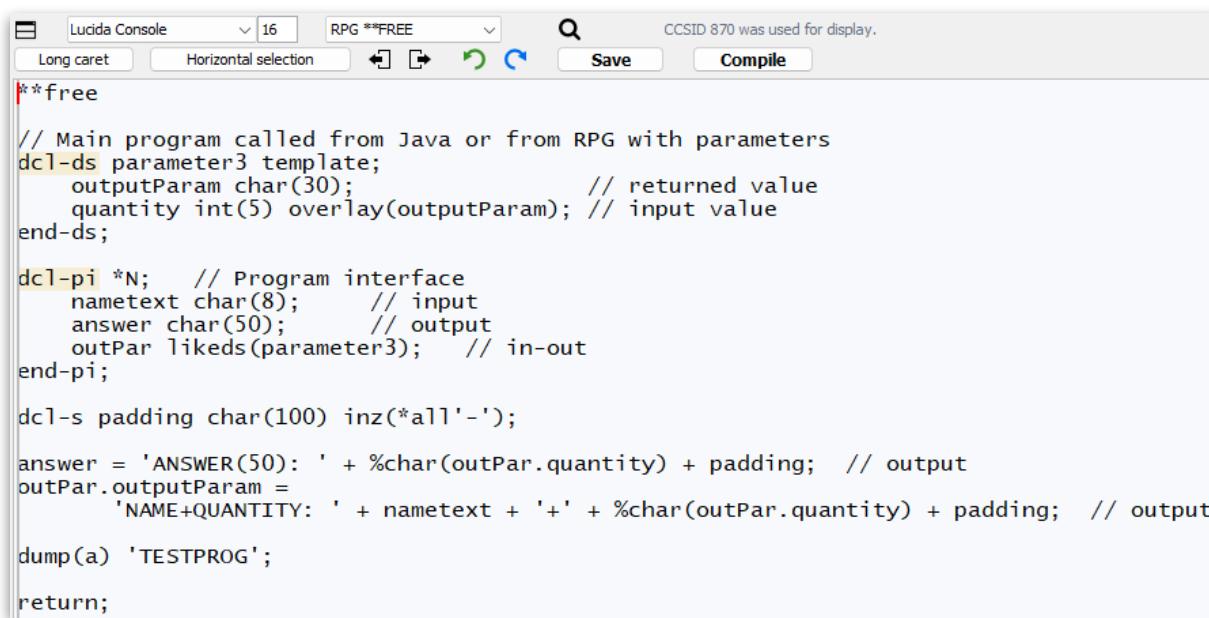
Zdrojový člen se v aplikaci chová podobně jako *soubor*.

Chování aplikace se řídí parametry uloženými v adresářích "paramfiles" a "workfiles".

Editace souborů

Texové soubory, zejména programové v IBM i a v PC lze zobrazit a editovat výběrem příkazu *Edit z kontextové nabídky* nebo *poklepáním* na jméno souboru v zobrazeném *stromu*.

Obsah souboru se zobrazí v samostatném okně s informací o *znakové sadě* textu v souboru.



The screenshot shows a window titled "Lucida Console" with a font size of 16 and "RPG **FREE" selected. The status bar at the top right says "CCSID 870 was used for display". The menu bar includes "File", "Edit", "View", "Format", "Search", "Tools", and "Help". Below the menu is a toolbar with icons for "Long caret", "Horizontal selection", "Find", "Save", and "Compile". The main text area contains the following RPG code:

```
**free

// Main program called from Java or from RPG with parameters
dcl-ds parameter3 template;
    outputParam char(30); // returned value
    quantity int(5) overlay(outputParam); // input value
end-ds;

dcl-pi *N; // Program interface
    nametext char(8); // input
    answer char(50); // output
    outPar likeds(parameter3); // in-out
end-pi;

dcl-s padding char(100) inz(*all'-');

answer = 'ANSWER(50): ' + %char(outPar.quantity) + padding; // output
outPar.outputParam =
    'NAME+QUANTITY: ' + nametext + '+' + %char(outPar.quantity) + padding; // output

dump(a) 'TESTPROG';

return;
```

V horní části okna jsou ovládací komponenty:

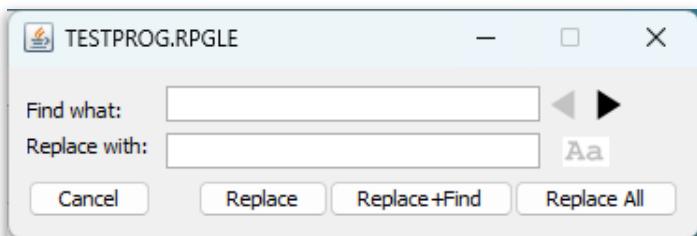
- *Delení/spojení* - střídavé tlačítka rozdělující text editoru na horní a spodní oblast vodorovnou čárou nebo obě oblasti spojí. Ve spodní oblasti je kopie textu editoru. Všechny změny provedené v jedné oblasti se automaticky promítají do druhé oblasti.
- *Lucida Console aj.* – z rozbalovacího seznamu vyberte písмо pro text v editoru.
- *RPG **FREE etc.* – z rozbalovacího seznamu vyberte programovací jazyk 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* – tlačítka posune vybraný text o jednu pozici vlevo, tlačítka 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.
- *Compile* – vyvolá okno pro komplikaci (viz [níže](#)), je-li editovaný soubor zdrojový člen nebo IFS soubor. Toto tlačítko se nezobrazuje u PC souborů.

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

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

Hledání textu

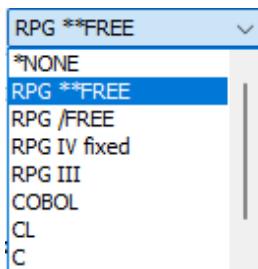
Klepnutí na lupu nebo stisk klávesové kombinace Ctrl F (Cmd F v systému macOS) vyvolá následující okno.



- *Find what* – zapište 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 zeštědne. Stejnou funkci mají klávesové zkratky Ctrl ↑ (předchozí) a Ctrl ↓ (další).
- *Replace with* – nahradní 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 nahradním textem.
- *Replace+Find* – nahradí právě nalezený text nahradním textem a nalezne *následující shodný text*.
- *Replace All* – nahradí *všechny* shodné texty nahradním textem.

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

Zvýraznění bloků



Rozbalovací seznam v dialogovém okně obsahuje značky programovacích jazyků, jejichž složené příkazy (bloky) mají být zvýrazněny. Seznam obsahuje tyto zápisy:

***NONE** – nic se nezvýrazňuje,

RPG **FREE – verze RPG se zcela volným zápisem příkazů,

RPG /FREE – verze RPG dovolující zápis příkazů ve formulářích i ve volném tvaru,

RPG IV fixed – verze RPG dovolující zápis příkazů v rozšířených formulářích,

RPG III – verze RPG/400 dovolující zápis v tradičních formulářích,

COBOL – jazyk COBOL,

CL – jazyk Control language,

C – jazyk C,

SQL – skript jazyka SQL.

Způsob zvýrazňování složených příkazů použitý v této aplikaci může mít některé nežádoucí účinky, protože krátké sekvence písmen – zvláště IF, DO, FOR, se mohou vyskytovat i jinde v textu. To však může být pro příslušné programovací jazyky přijatelné.

Blok RPG programu s volbou *RPG /FREE* nebo *RPG **FREE*:

```
dcl-PROC VRATIT_CENU;
    dcl-ds cenik_ds extname('*LIBL/CENIKP') qualified end-ds;

    dcl-pi *n packed(10: 2);
        material like(cenik_ds.MATER) CONST; // nebo VALUE
    end-pi;

    exec SQL select CENA into :cenik_ds.CENA from CENIKP
        where MATER = :material;
    if sqlstate >= '02000';
        cenik_ds.CENA = -1;
    endif;

    return cenik_ds.CENA;
end-PROC;
```

Blok CL programu s volbou *CL*:

```
DoUntil ('0') /* Execute at least once */
    Rcvf opnid(FILES) /* Read from the list of file objects */
    Monmsg msgid(CPF0864) exec(Leave) /* EOF - quit this loop */

    If (&FILES_ODOBAT = 'PF') Do /* If a Physical file then ok */
        /* Generate a file with a list of members of this file */
        Dspfd file(&Library/&FILES_ODOBNAME) Type(*MBRLIST) +
            output(*OUTFILE) outfile(QTEMP/MEMBERS) +
            outmbr(*FIRST *ADD)
    EndDo /* If */
EndDo /* Do Until */
```

COBOL program:

```
.....*A+++B+++++++++++++++++++++Pgm-id++
IDENTIFICATION DIVISION.
PROGRAM-ID. Conditions.
AUTHOR. Michael Coughlan.
* An example program demonstrating the use of
* condition names (level 88's).
* The EVALUATE and PERFORM verbs are also used.

DATA DIVISION.
WORKING-STORAGE SECTION.
01 Char          PIC X.
88 Vowel         VALUE "a", "e", "i", "o", "u".
88 Consonant     VALUE "b", "c", "d", "f", "g", "h"
                  "j" THRU "n", "p" THRU "t", "v" THRU "z".
88 Digit          VALUE "0" THRU "9".
88 ValidCharacter VALUE "a" THRU "z", "0" THRU "9".

PROCEDURE DIVISION.
Begin.
    DISPLAY "Enter lower case character or digit. No data ends.".
    ACCEPT Char.
    PERFORM UNTIL NOT ValidCharacter
        EVALUATE TRUE
        WHEN Vowel DISPLAY "The letter " Char " is a vowel."
        WHEN Consonant DISPLAY "The letter " Char " is a consonant."
        WHEN Digit DISPLAY Char " is a digit."
        WHEN OTHER DISPLAY "problems found"
    END-EVALUATE
    ACCEPT Char
END-PERFORM
STOP RUN.
```

Posouvání vybraného textu

Tlačítka a posouvají vybraný text o jednu pozici vlevo a vpravo. Stejnou funkci vykonají klávesové zkratky **Ctrl ←** a **Ctrl →**. V systému macOS se použije klávesa **Cmd** místo Ctrl. Výsledky posuvů mohou být odvolány a přivolány.

Horizontal selection

Vybrané řádky se posunou *vpravo* zároveň s nevybraným zbytkem posledního řádku.

Vybrané řádky se posunou *vlevo*, jestliže všechny obsahují vlevo aspoň jednu mezeru.

Příklad

Horizontální výběr přes 3 řádky:

The screenshot shows an RPG editor interface with the following details:

- Top bar: Lucida Console, 14, RPG **FREE, Save, Compile.
- Message: CCSID 870 was used for display.
- Toolbar: Long caret, Horizontal selection, , , , .
- Text area:

```
**free
// Main program called from Java or from RPG with parameters
dcl-ds parameter3 template;
    outputParam char(30);          // returned value
    quantity int(5) overlay(outputParam); // input value
end-ds;

dcl-pi *N; // Program interface
    nametext char(8); // input
    answer char(50); // output
    outPar likeds(parameter3); // in-out
end-pi;

dcl-s padding char(100) inz(*all'-'');
```

A horizontal selection is highlighted around the second and third lines of code, specifically around the `dcl-pi` and `nametext` declarations.

Po 5 stiscích tlačítka byl vybraný text posunut o 5 pozic vpravo:

The screenshot shows the same RPG editor interface after pressing the right arrow key 5 times. The horizontal selection has moved 5 positions to the right:

- Top bar: Lucida Console, 14, RPG **FREE, Save!, Compile.
- Text area:

```
**free
// Main program called from Java or from RPG with parameters
dcl-ds parameter3 template;
    outputParam char(30);          // returned value
    quantity int(5) overlay(outputParam); // input value
end-ds;

dcl-pi *N; // Program interface
    nametext char(8); // input
    answer char(50); // output
    outPar likeds(parameter3); // in-out
end-pi;

dcl-s padding char(100) inz(*all'-'');
```

The selection is now positioned between the `nametext` and `answer` declarations in the second line of code.

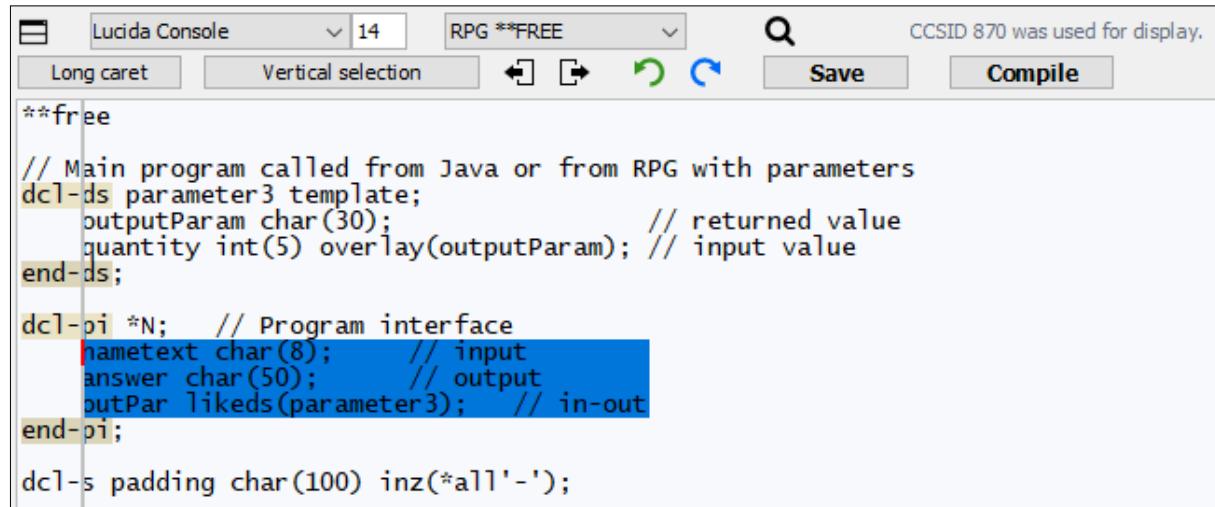
Vertical selection

Vybraný obdélník se posune vlevo o jednu pozici, jestliže zbývá alespoň jeden sloupec mezer v nevybrané části nalevo od obdélníka. Obdélník přepíše sloupce nalevo a nechá za sebou sloupec mezer (vpravo).

Vybraný obdélník se posune vpravo o jednu pozici zároveň se zbytkem řádků a nechá za sebou sloupec mezer (vlevo).

Příklad

Při vertikálním výběru byl vybrán obdélník textu:



The screenshot shows a text editor window with the following configuration:

- Title bar: Lucida Console, 14, RPG **FREE, CCSID 870 was used for display.
- Toolbar buttons: Long caret, Vertical selection, Save, Compile.
- Text area:

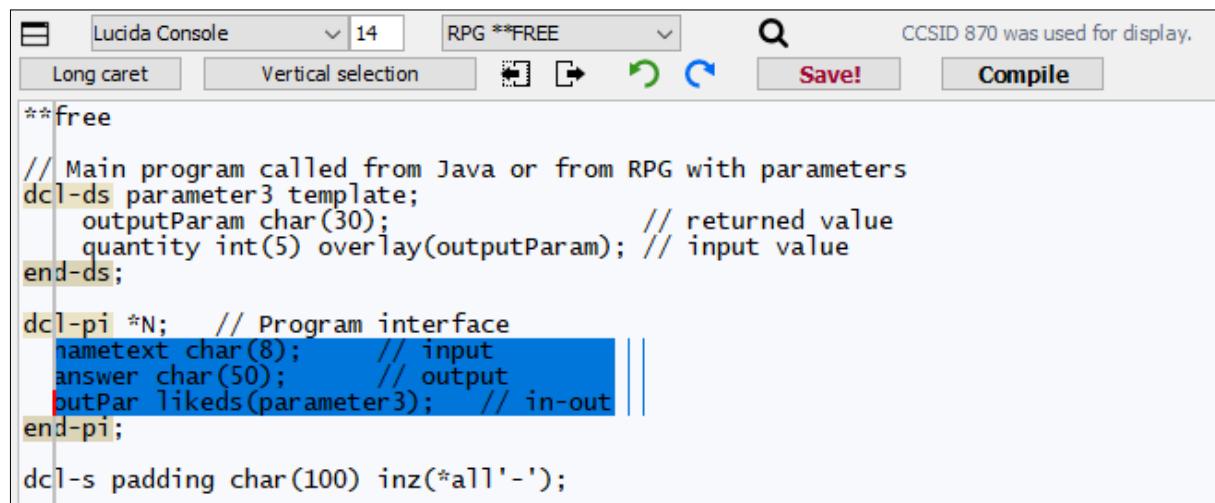
```
**free
// Main program called from Java or from RPG with parameters
dcl-ds parameter3 template;
    outputParam char(30);          // returned value
    quantity int(5) overlay(outputParam); // input value
end-ds;

dcl-pi *N;      // Program interface
    nametext char(8);        // input
    answer char(50);         // output
    putPar likeds(parameter3); // in-out
end-pi;

dcl-s padding char(100) inz(*all'-'');
```

A vertical selection box is drawn around the first two lines of code: `**free` and `// Main program called from Java or from RPG with parameters`. The selection box is highlighted with a yellow border.

Po 2 stiscích tlačítka ← byl obdélník posunut o 2 pozice vlevo:



The screenshot shows the same text editor window after performing a horizontal move:

- Title bar: Lucida Console, 14, RPG **FREE, CCSID 870 was used for display.
- Toolbar buttons: Long caret, Vertical selection, Save!, Compile.
- Text area:

```
**free
// Main program called from Java or from RPG with parameters
dcl-ds parameter3 template;
    outputParam char(30);          // returned value
    quantity int(5) overlay(outputParam); // input value
end-ds;

dcl-pi *N;      // Program interface
    nametext char(8);        // input
    answer char(50);         // output
    putPar likeds(parameter3); // in-out
end-pi;

dcl-s padding char(100) inz(*all'-'');
```

The previously selected lines have moved two positions to the left. The selection box is now positioned around the second and third lines of code: `dcl-ds parameter3 template;` and `outputParam char(30);`. The selection box is highlighted with a yellow border.

Copy, cut and paste selected text

Ke kopírování, vyjímání a vkládání se používají běžné klávesové zkratky **Ctrl-C**, **Ctrl-X**, **Ctrl-V** (copy, cut, paste). V systému macOS se použije klávesa **Cmd** místo Ctrl.

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 byt v oblasti editoru, anebo někde jinde v PC.

Výsledky těchto operací mohou být odvolány a přivolány.

Horizontal selection

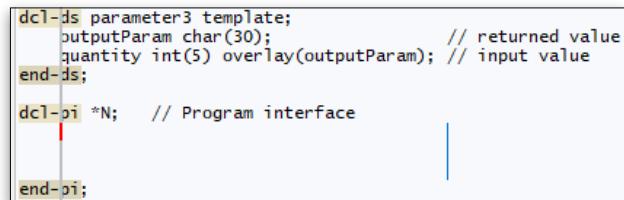
Všechny tyto operace fungují jako v PC.

Vertical selection

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

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

Například obdélník z předchozího obrázku byl vyňat operací Cut:



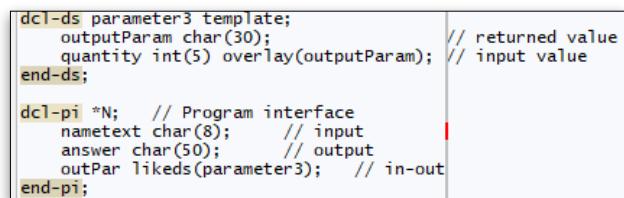
```
dcl-ds parameter3 template;
    outputParam char(30);           // returned value
    quantity int(5) overlay(outputParam); // input value
end-ds;

dcl-pi *N;      // Program interface

end-pi;
```

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

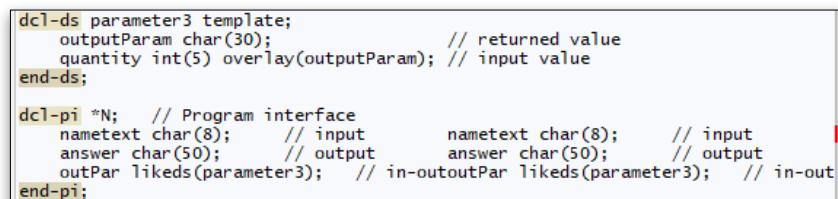
Operace *Paste* *přepíše* oblast v editoru *vpravo* a *dolů* od pozice zvolené ukazatelem. Může to být i původně vybraná oblast (operací *Copy* nebo *Cut*).



```
dcl-ds parameter3 template;
    outputParam char(30);           // returned value
    quantity int(5) overlay(outputParam); // input value
end-ds;

dcl-pi *N;      // Program interface
    nametext char(8);           // input
    answer char(50);            // output
    outPar likeds(parameter3);  // in-out
end-pi;
```

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



```
dcl-ds parameter3 template;
    outputParam char(30);           // returned value
    quantity int(5) overlay(outputParam); // input value
end-ds;

dcl-pi *N;      // Program interface
    nametext char(8);           // input
    answer char(50);            // output
    outPar likeds(parameter3);  // in-out
    nametext char(8);           // input
    answer char(50);            // output
    outPar likeds(parameter3);  // in-out
end-pi;
```

Je-li oblast editoru kratší než vkládaný obdélník, doplní se potřebné prázdné řádky a vkládaná data se do nich vloží (za konec oblasti).

```
dcl-ds parameter3 template;
    outputParam char(30);           // returned value
    quantity int(5) overlay(outputParam); // input value
end-ds;

dcl-pi *N;      // Program interface
    nametext char(8);        // input
    answer char(50);         // output
    outPar likeds(parameter3); // in-out
end-pi;

dcl-s padding char(100) inz(*all'-');

answer = 'ANSWER(50): ' + %char(outPar.quantity) + padding; // output
outPar.outputParam =
    'NAME+QUANTITY: ' + nametext + '+' + %char(outPar.quantity) + padding; // output

dump(a) 'TESTPROG';

return;
nametext char(8);      // input
answer char(50);       // output
outPar likeds(parameter3); // in-out
```

Pomůcka pro jazyky založené na formulářích

Jazyky RPG III, starší verze RPG IV, COBOL a DDS používají formuláře pro zápis příkazů. V nabídce *Help* jsou k dispozici *záhlaví formulářů*, která lze zkopirovat a umístit do editovaného textu na vhodná místa jako komentář a vést tak programátora, aby programové specifikace zapisoval do správných sloupců. Například lze na vhodné místo vložit formát popisu souborů v RPG III programu

```
.....F*filenameIPEAF....RlenLK1AI0vKlocEDevice+.....KOptionEntry+A....U1.....
```

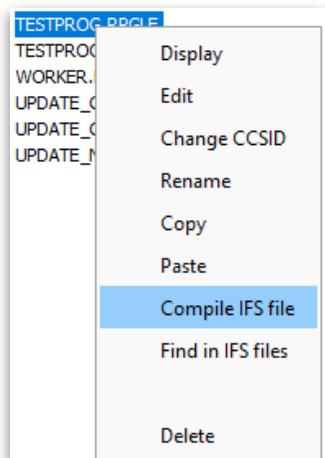
nebo formát DSPF v popisu DDS

```
.....A*N01N02N03T.Name+++++RLen++TDpBLinPosFunctions++++++++++++++++++++++
```

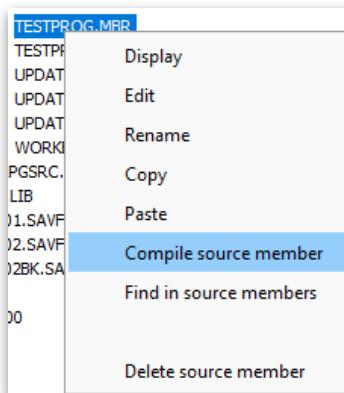
Kompilace

Kontextová nabídka u některých souborů v IBM i obsahuje příkaz ke komplikaci. U IFS souborů je to příkaz *Compile IFS file*, u zdrojových členů *Compile source member*.

IFS file



Source member



Jiný způsob zahájení komplikace je stisk tlačítka *Compile* v textovém editoru.

Poznámka: Pojem *kompilace* znamená vytváření objektů příslušnými příkazy. Většinou začínají písmeny CRT, např. CRTBNDRPG k vytvoření RPG programu. Výjimkou je vytvoření objektu z SQL skriptu, kde vytvářejícím příkazem je RUNSQLSTM.

Zobrazí se kompilační okno.



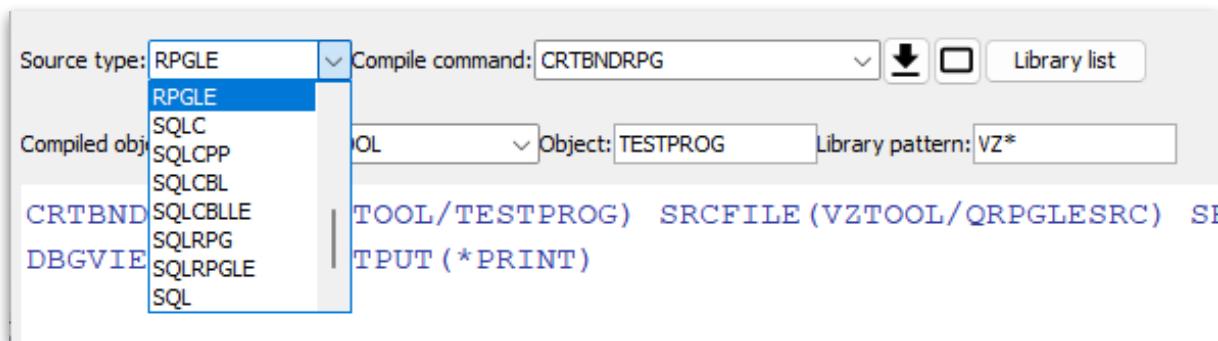
Okno je členěno na několik oddílů..

1. Zobrazuje *cestu* ke zdrojovému členu nebo IFS souboru (tučným písmem).
2. *Parametry vstupu* do komplikace - typ zdroje, jméno příkazu a volba serznamu knihoven.
3. Údaje potřebné k identifikaci *výstupu* z komplikace.
 - Jméno cílové *knihovny*.
 - Jméno cílového *objektu*.
 - Výběrový vzorek k omezení seznamu knihoven v rozbalovacím seznamu – lze jej zapsat do vstupního pole.
4. Zobrazuje aktuální text kompilačního příkazu.
5. Tlačítka:
 - Cancel* – zruší práci. Práci lze zrušit také uzávěrem okna a také *klávesou ESC*.
 - Perform command* – provede (spustí) kompilační příkaz,
 - Last spooled file* – zobrazí naposledy vytvořený tiskový soubor pro současného uživatele.
 - Spooled files* – získá seznam tiskových souborů,
 - Job log* – tiskne protokol úlohy,
 - Clear messages* – smaže všechny zprávy.
6. Prostor pro zprávy.

Poznámka: Kompilační příkaz se normálně spustí tlačítkem *Perform command*. Je ale také možné příkaz modifikovat nebo přepsat zcela jiným příkazem. Lze zapsat libovolný CL příkaz a spustit jej. Tak třeba lze příkazem CALL spustit neinterakční program. Výsledky pak mohou být třeba ve fyzickém nebo tiskovém souboru, anebo v protokolu úlohy.

Source type

Zdrojové členy



Při prvním zobrazení kompilačního okna je *Typ zdrojového členu* automaticky dosazen. Zároveň je určen kompilační příkaz. Je ale možné typ opravit buď výběrem z rozbalovacího seznamu nebo přímým zápisem do vstupního pole. Takto změněný typ se použije k určení příslušného kompilačního příkazu. Skutečný typ zdrojového členu se však nemění.

Kompilovat lze programy, a soubory s popisy dat a tabulek následujících zdrojových typů:

CLLE, CLP,
RPG, RPGLE, SQLRPG, SQLRPGLE,
CBL, CBLLE, SQLCBL, SQLCBLLE,
C, CPP, SQLC, SQLCPP,
CMD,
DSPF, LF, PF, PRTF,
SQL,
TBL

U zdrojových členů v zobrazeném stromu jsou zobrazeny skutečné zdrojové typy jako přípony.

Důležité: Změna typu zdrojového členu je možná jen CL příkazem WRKMBRPDM.

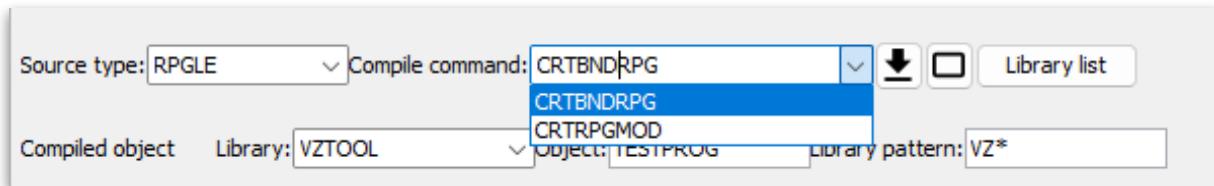
IFS soubory

U IFS souborů musí být výslovně zapsány výše uvedené přípony (za tečkou), nemusí být ale psány velkými písmeny. Z IFS souborů ale *nelze kompilovat* zdroje následujících typů:

CLLE, CLP, CMD, RPG, CBL, SQLRPG, SQLCBL
DSPF, LF, PF, PRTF,
TBL

Compile command

CL příkaz komplikace lze buď ponechat beze změny, nebo je nutno jej vybrat z rozbalovacího seznamu, či zapsat do vstupního pole. Některé zdrojové typy se totiž mohou komplikovat jako *program* nebo jako *modul*, nebo jako *servisní program*.



Program nebo modul lze vytvářet ze zdrojových typů ILE: CLLE, RPGLE, CBLLE, C, CPP, SQLRPGLE, SQLCBLLE, SSQLC a SQLCPP.

Servisní program lze vytvářet ze zdrojových typů s SQL příkazy: SQLRPGLE, SQLCBLLE a SSQLC.

Uchování atributů komplikace

Dvě tlačítka umožňují ukládat nebo vyčistit *atributy komplikace* (zdrojový typ, kompilační příkaz, cílová knihovna, cílový objekt) vybrané z rozbalovacích seznamů a textového pole.

Tlačítko "Ukládat", které má dvě střídavé ikony a , definuje jak se s vybranými hodnotami zachází.

Má-li tlačítko *černou* ikonu, hodnoty se *ukládají* a přiřazují ke každému komplikovanému souboru. Tak se *atributy komplikace* fixují jako uživatelsky definovaná předvolba. Objeví se v rozbalovacích seznamech pokaždé, když je soubor vybrán ke komplikaci, dokud je uživatel nezmění.

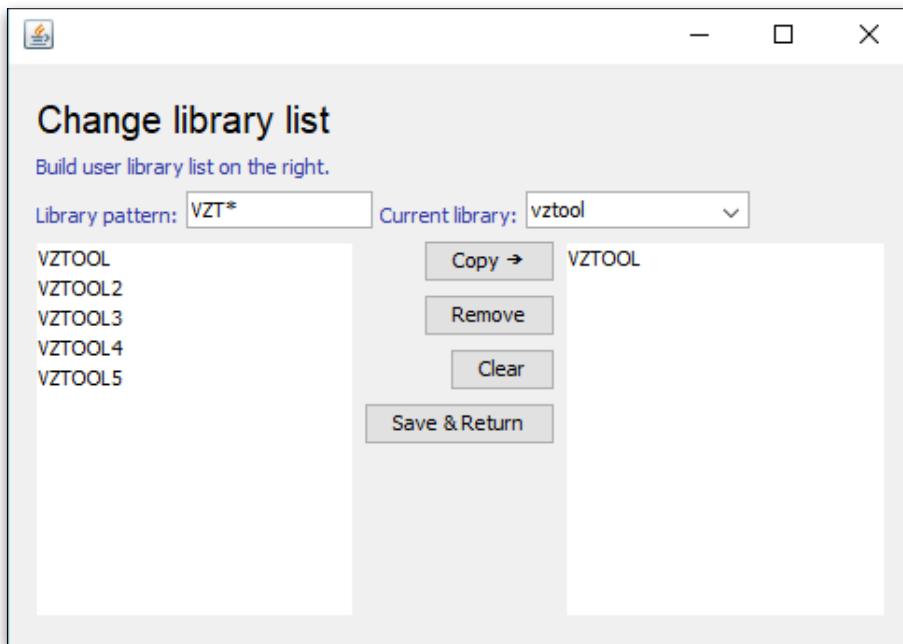
Má-li tlačítko *světle šedou* ikonu, hodnoty se *neukládají*.

Tlačítko "Vyčistit" s ikonou , je-li stisknuto, smaže uložené hodnoty pro všechny komplikované soubory.

Poznámka: Uložené hodnoty jsou zapsány v souboru *CompileAttributes.lib* v adresáři *workfiles*.

Change library list

Tlačítko zobrazí okno, kde lze nastavit uživatelský seznam knihoven (user library list) a běžnou knihovnu (current library).



Library pattern

V poli *Library pattern* lze zadat *vyhledávací vzorek* (viz kapitola *LIB, FILE, MBR*) a stiskem klávesy *Enter* získáme v levém rámečku seznam knihoven jejichž jména vyhovují tomuto vzorku. Je-li vstupní pole prázdné, bude v levém rámečku seznam všech knihoven.

Current library

V rozbalovacím seznamu *Current library* lze volit běžnou knihovnu ze stejného seznamu, který je zobrazen v levém rámečku, nebo lze zapsat jméno knihovny do vstupního pole rozbalovacího seznamu. Zvláštní položkou je symbol *CRTDFT, který znamená, že seznam knihoven úlohy (job library list) *neobsahuje* běžnou knihovnu (current library).

Vytváření uživatelského seznamu knihoven

Z levé strany na pravou lze přenášet označené knihovny buď metodou *drag and drop*, anebo stiskem tlačítka **Copy →**. Pravý rámeček představuje uživatelskou část seznamu knihoven v úloze (job).

Tlačítko *Remove* odstraní označené knihovny z pravého rámečku.

Tlačítko *Clear* vyčistí pravý rámeček.

Tlačítko *Save & return* uloží provedené změny.

Compiled object

Compiled object	Library: VZTOOL	Object: TESTPROG	Library pattern: VZTOOL
-----------------	-----------------	------------------	-------------------------

U komplikace *zdrojového členu* jsou předvolené parametry v tomto řádku odvozeny z *cesty* ke zdrojovému členu. Lze je změnit a potvrdit klávesou Enter.

U komplikace *IFS souboru* může být nutné tyto parametry změnit, protože z cesty k IFS souboru nelze cílovou knihovnu a objekt odvodit jako z cesty ke zdrojovému členu.

Předvolené jméno knihovny je převzato z první položky rozbalovacího seznamu. Předvolené jméno cílového objektu převzato z *cesty* a je zkráceno na 10 znaků.

Library

Knihovnu lze vybrat z rozbalovacího seznamu nebo zapsat do vstupního pole a stisknout klávesu Enter.

Object

Jméno objektu lze změnit ve vstupním poli a stisknout klávesu Enter.

Library pattern

Zápisem vzorku do tohoto vstupního pole a stiskem klávesy Enter se do rozbalovacího seznamu zapíší knihovny, jejichž jména vyhovují vzorku. Vzorek může obsahovat znaky * a ? (wildcards).

Perform command

Tlačítkem *Perform command* se spustí komplikace. O jejím výsledku se zobrazí zpráva ve spodní části okna, např.:

CPC0815 *COMPLETION: Program CHCKPROFIL created in library VZ_DTAARA.

Cause: See the compiler printout for a list of any messages. Recovery: If necessary, correct the errors.

Protokol o komplikaci (listing) se vytiskne do tiskového souboru QPRINT.

Spooled files

Tlačítko Spooled files zobrazí tabulkou tiskových souborů *současného uživatele*. Výběr, zobrazení, kopírování a výmaz tiskových souborů se provádí stejně jako u objektů typu OUTQ (viz [Tiskové soubory \(spooled files\)](#) výše).

Okno obsahuje navíc tlačítko *Refresh*, jímž lze obnovit tabulku tiskových souborů bez nového stisku tlačítka Spooled files.

File name	File num.	Job name	User	Job num.	Date	Time	
File name	File num.	Job name	User	Job num.	Date	Time	Output queue
CENY2	1	QPRTJOB	VZUPKA	029937	1170311	094622	QGPL/QPRINT

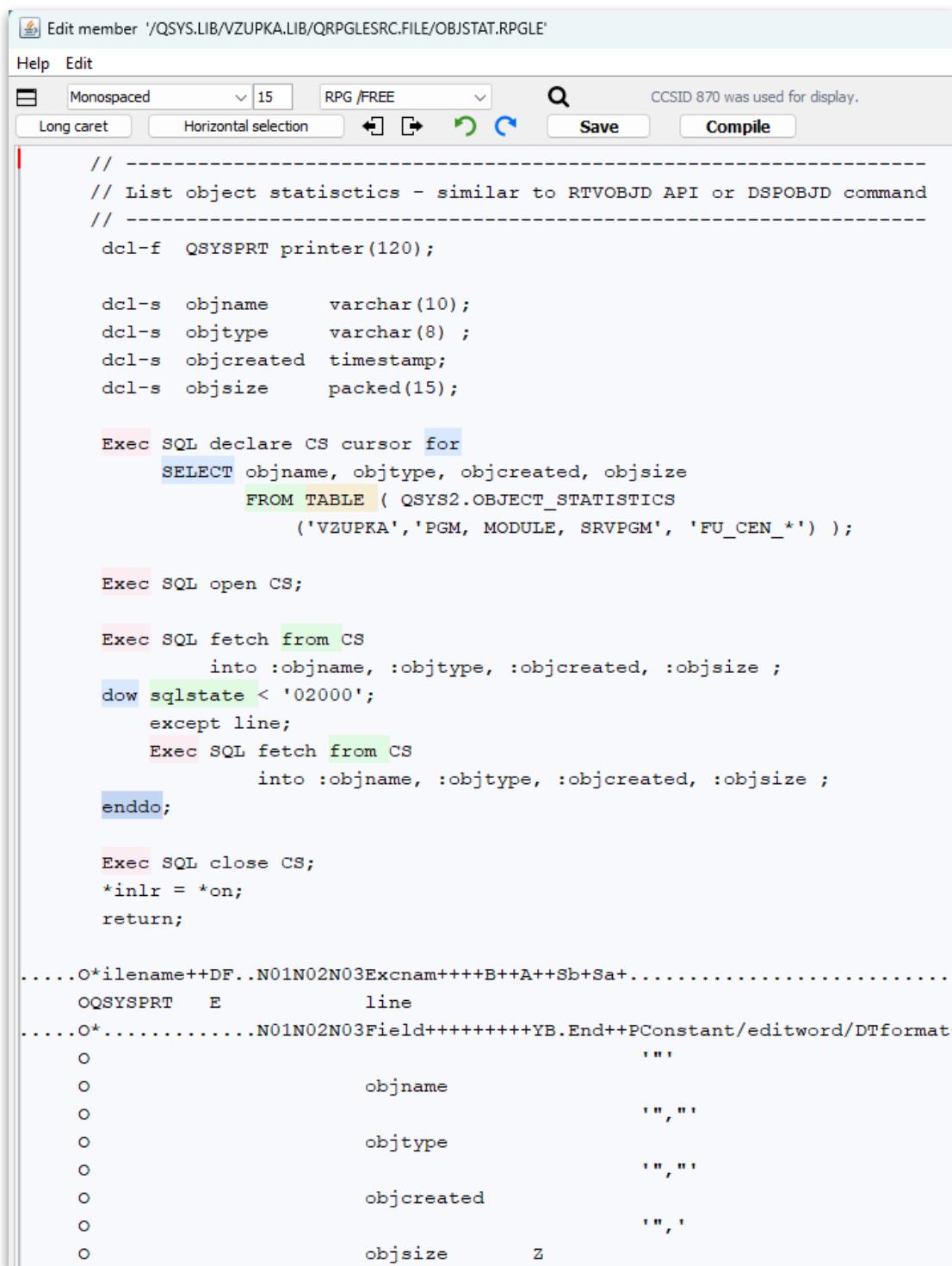
Klávesa ESC odstraní okno s tiskovými soubory.

Job log

Tlačítko *Job log* vytiskne momentální obsah protokolu úlohy do tiskového souboru QEZJOBLOG. Pomocí tlačítka Spooled files lze obsah protokolu nalézt a zobrazit jej podobně jako jiné textové soubory.

Příklad komplikace

Program OBJSTAT získá údaje z tabulkové funkce QSYS2.OBJECT_STATISTICS a vypíše je do tiskového souboru QSYSPRT.



The screenshot shows an RPGLE editor window titled "Edit member '/QSYS.LIB/VZUPKA.LIB/QRPGLESRC.FILE/OBJSTAT.RPGL'". The interface includes a toolbar with "Help", "Edit", "Monospaced", "15", "RPG /FREE", "Save", and "Compile" buttons. A status bar at the top right indicates "CCSID 870 was used for display". The code itself is written in RPGLE and performs the following tasks:

- It declares a cursor named CS that selects object statistics from the QSYS2.OBJECT_STATISTICS table for objects in the 'VZUPKA' library.
- It opens the cursor and fetches rows until a SQL error occurs.
- For each row, it prints the object name, type, creation timestamp, and size.
- Finally, it prints the field descriptions for the output fields: objname, objtype, objcreated, and objsize.

```
// -----
// List object staticctics - similar to RTVOBJD API or DSPOBJD command
// -----
dcl-f  QSYSPRT printer(120);

dcl-s  objname      varchar(10);
dcl-s  objtype       varchar(8) ;
dcl-s  objcreated    timestamp;
dcl-s  objsize       packed(15);

Exec SQL declare CS cursor for
      SELECT objname, objtype, objcreated, objsize
        FROM TABLE ( QSYS2.OBJECT_STATISTICS
                  ('VZUPKA','PGM, MODULE, SRVPGM', 'FU_CEN_*') ) ;

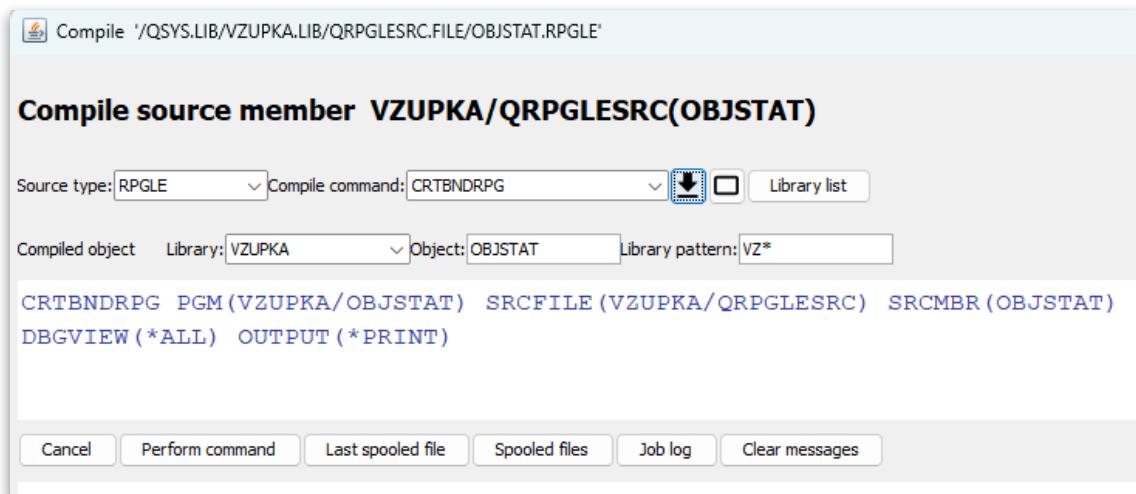
Exec SQL open CS;

Exec SQL fetch from CS
      into :objname, :objtype, :objcreated, :objsize ;
dow sqlstate < '02000';
  except line;
  Exec SQL fetch from CS
      into :objname, :objtype, :objcreated, :objsize ;
enddo;

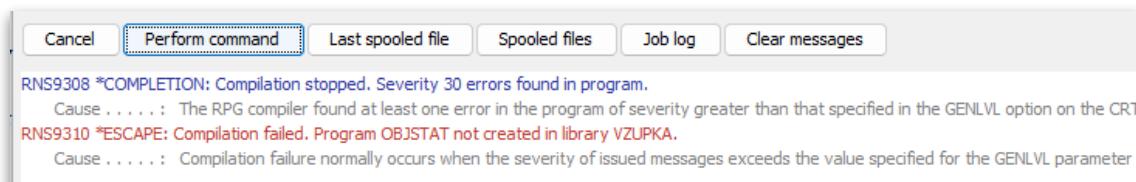
Exec SQL close CS;
*inlr = *on;
return;

....O*filename++DF..N01N02N03Excnam++++B++A++Sb+Sa+.....
  OQSYSPRT   E           line
....O*.....N01N02N03Field++++++YB.End++PConstant/editword/DTformat
  O                      ""
  O                      objname
  O                      ""
  O                      objtype
  O                      ""
  O                      objcreated
  O                      ""
  O                      objsize      Z
```

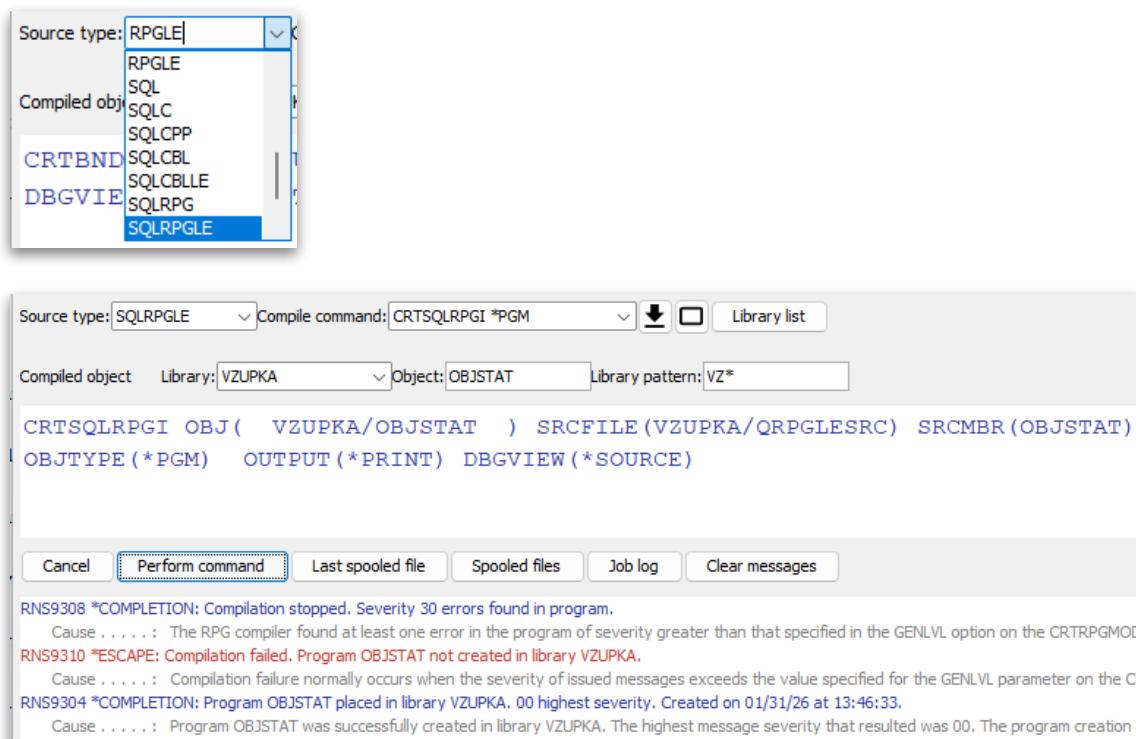
Stisk tlačítka *Compile* ukáže okno s kompilačním příkazem a předvolenými parametry.



Stiskem tlačítka *Perform command* se spustí komplikace a ukáže se výsledek.



Ohlásila se chyba, protože předvolený typ RPGLE převzatý ze skutečného typu zdrojového členu byl nesprávný. Program totiž obsahuje příkazy SQL. Proto změníme zdrojový typ na SQLRPGLE a nová komplikace tlačítkem *Perform command* už vytvoří program.

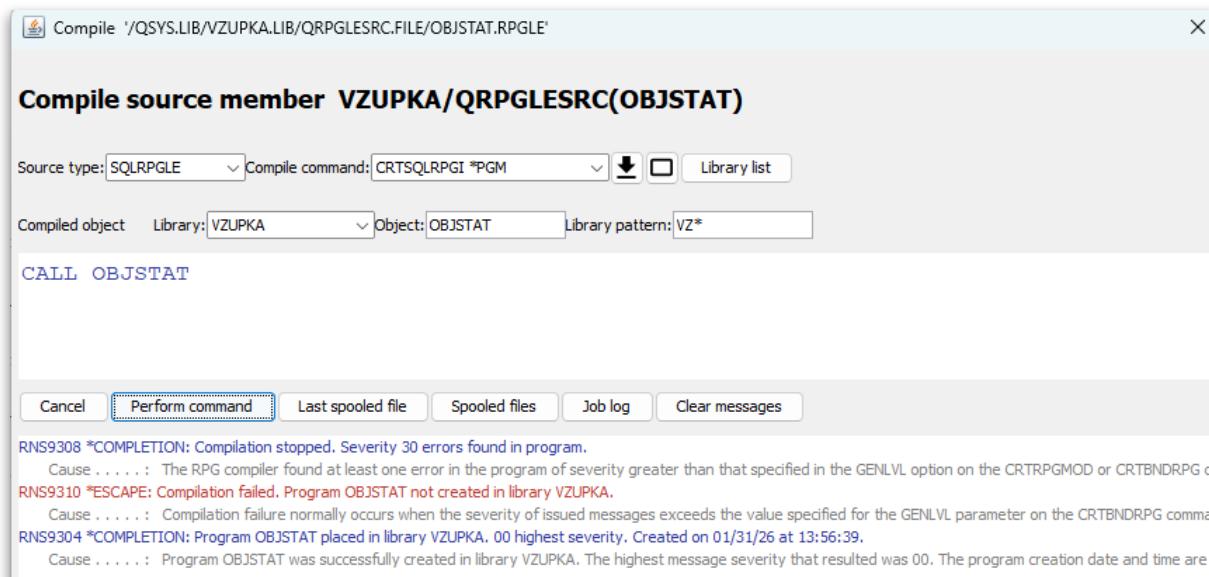


Když budeme program třeba několikrát (i mnohokrát) upravovat, nemusíme tento postup opakovat. To zařídí tlačítka a , které jsouc přepnuto do černé barvy, uloží kompilační příkaz tohoto programu do položky interního adresáře *workfiles*. Pamatuje si ho tak dlouho, dokud nestiskneme tlačítka , které smaže všechny položky z adresáře.

Poznámka: Nehledě na typ RPGLE ve vstupním poli kompilační příkaz CRTSQLRPGI stále zůstává v platnosti.

Spuštění CL příkazu

Správně kompliovaný program můžeme spustit tak, že kompilační příkaz smažeme a zapíšeme příkaz CALL OBJSTAT, načež stiskneme tlačítka *Perform command*.



Protože byl správně nastaven seznam knihoven a příkaz byl správně zapsán, nezobrazí se žádná další zpráva.

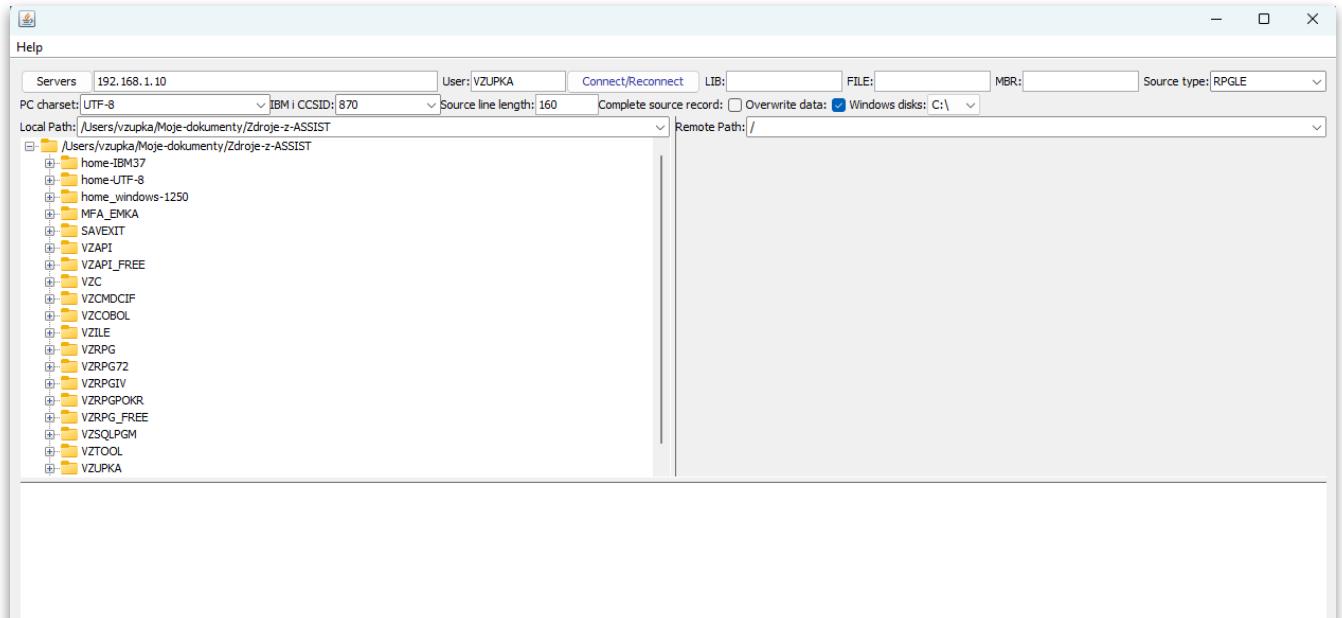
Tlačítkem *Last Spooled File* zobrazíme data tiskového souboru QSYSPRT. Nebo po stisku tlačítka *Spooled files* najdeme tento soubor v seznamu a zobrazíme jeho data.

Find:	Data	CCSID 870 was used for display.
"FU_CEN_C "	"*PGM " , "2025-03-29-11.30.31.000000"	176128
"FU_CEN_C "	"*MODULE " , "2025-03-29-11.21.46.000000"	135168
"FU_CEN_R "	"*MODULE " , "2025-03-29-11.30.12.000000"	163840
"FU_CEN_R "	"*SRVPGM " , "2025-03-29-11.30.25.000000"	233472

Parametry aplikace

Po startu aplikace se zobrazí hlavní okno, v němž levá polovina zobrazuje strom představující souborový systém PC. Pravá polovina je zatím prázdná, zaplní se až po připojení k systému IBM i (pomocí tlačítka *Connect/Reconnect*).

Horní část obrazovky obsahuje parametry, které ovlivňují chování aplikace. Mají formu vstupních textových polí, rozbalovacích seznamů a zaškrťvacích políček.



IBM i server a uživatel

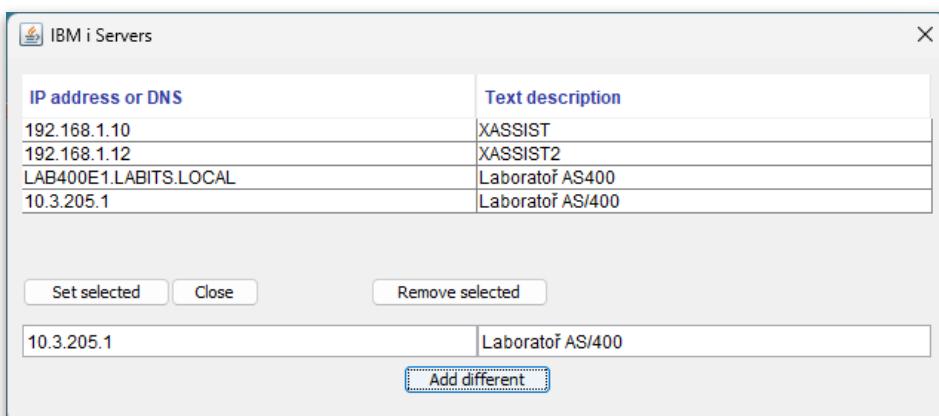
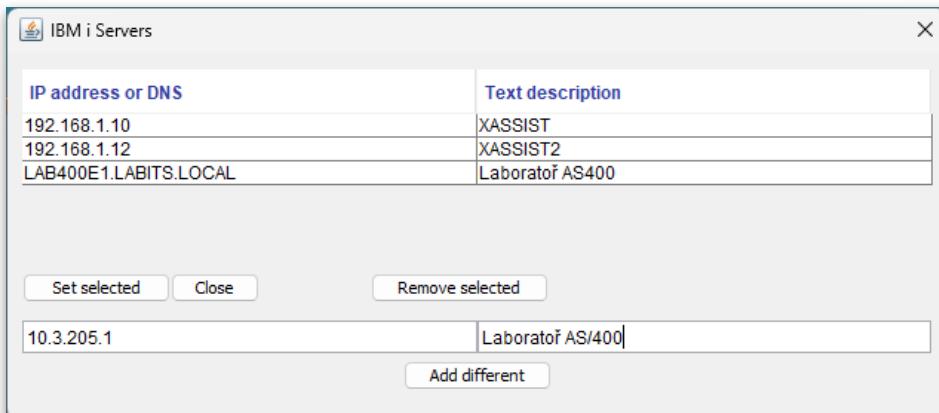
Servers	192.168.1.10	User:	VZUPKA	Connect/Reconnect
---------	--------------	-------	--------	-------------------

Uživatel musí do vstupního pole *User* zadat jméno uživatele. Adresu serveru IBM i může zadat ručně do vstupního pole nebo pomocí tlačítka *Servers*. Pak se připojí k serveru stiskem tlačítka *Connect/Reconnect*. Ostatní parametry může upravit později.

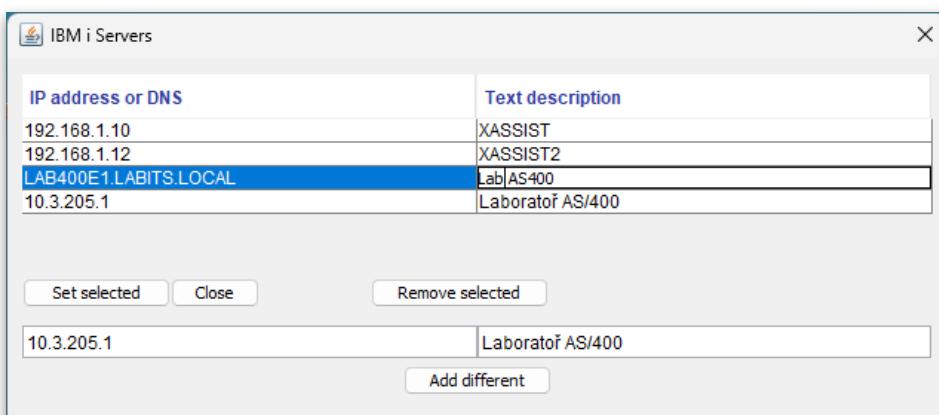
Servers

Toto tlačítko zobrazí okno k údržbě seznamu IBM i serverů k výběru pro připojení.

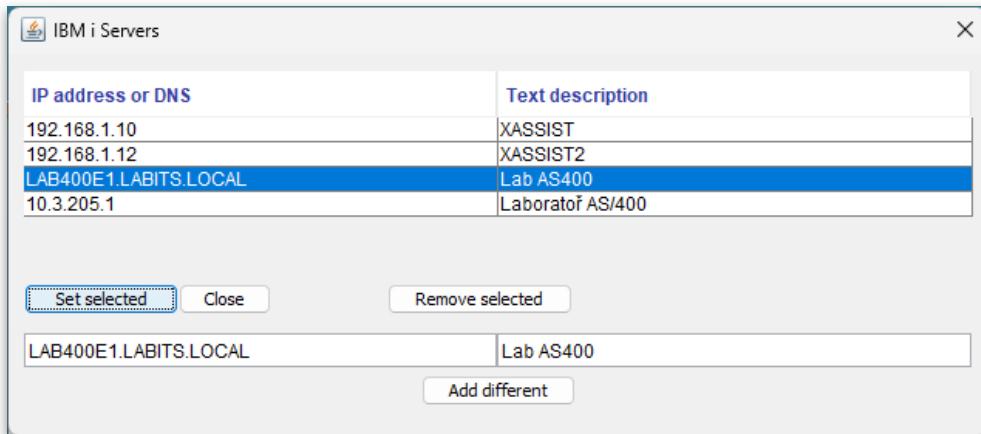
- Tlačítkem *Add different* se do seznamu přidá adresa a popis nového serveru ze dvou spodních vstupních polí. Obě hodnoty nesmí být prázdné a nesmí už být v seznamu obsaženy.



Ve vybraném řádku lze měnit každou buňku jako ve vstupním poli poklepáním, zápisem dat a stiskem klávesy *Enter*:



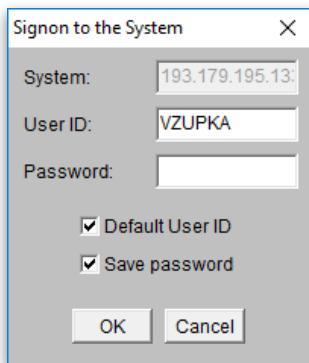
- Tlačítkem *Set selected* se server vybraný ze seznamu zkopiřuje do obou vstupních polí a také do vstupního pole v hlavním okně. Tím se nastaví adresa k připojení.



- Tlačítko *Remove selected* odebera vybraný server ze seznamu, ale adresa k připojení se nemění.

Connect/Reconnect

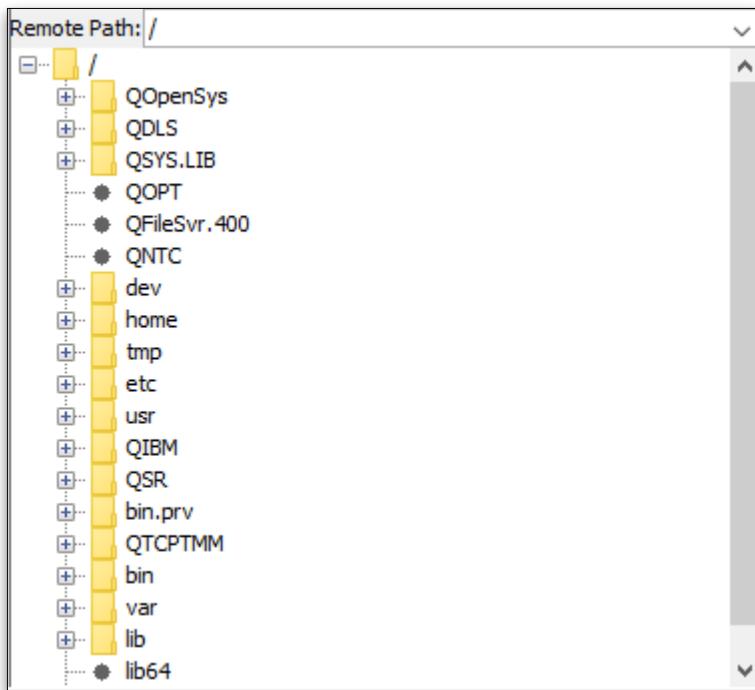
Stiskem tlačítka *Connect/Reconnect* se zobrazí dialog *Signon to the System*, kde uživatel zapíše heslo.



Přihlašovací proces může trvat déle, podle rychlosti připojení. Při dalším stisku tlačítka se server znova připojí. Nové připojení může být bez dialogu a trvat kratší dobu, jestliže uživatel nezměnil adresu serveru.

Uživatel může připojit jiný server, změní-li jeho adresu (třeba pomocí tlačítka *Servers*) a stiskne tlačítko *Connect/Reconnect* nebo klávesu *Enter*.

Po úspěšném připojení se v pravé polovině okna zobrazí strom představující souborový systém serveru IBM i. Ve stromu jsou zobrazeny všechny adresáře systému IFS, jestliže ve vstupním poli *Remote Path* je obsažen symbol kořene souborového systému IFS (znak /).



Vstupní pole LIB, FILE, MBR

Tato vstupní pole umožňují vybrat objekty uvnitř uzlu /QSYS.LIB (systémová knihovna). Uživatel může vybrat určité knihovny, soubory nebo členy zápisem vyhledávacího vzorku do odpovídajícího pole. Výběr může zahájit stiskem klávesy *Enter* nebo klepnutím na tlačítko *Connect/Reconnect*.

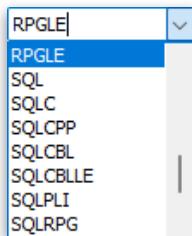
Vyhledávací vzorek může kromě normálních znaků obsahovat hvězdičku (*) a otazník (?).

- Znak * ignoruje nula nebo více znaků ve jméně objektu.
- Znak ? ignoruje jeden znak v dané pozici jména objektu.

Například zápis vzorku V?T* do pole vybere jména začínající písmenem V, obsahující libovolný znak ve druhé pozici, písmeno T ve třetí pozici a libovolné znaky ve zbývajících pozicích.

Prázdné pole vybírá všechna jména. Zápis přesného jména objektu vybere tento konkrétní objekt.

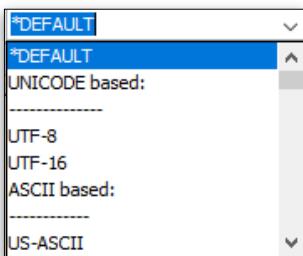
IBM i source type



Tato rozevírací nabídka se použije pro určení typu členu zdrojového fyzického souboru při jeho vytváření. Uživatel může zvolit typ z této rozevírací nabídky, nebo zapsat typ přímo do vstupního pole. Vytvoření členu se provádí volbou příkazu *New source physical file* v kontextové nabídce na uzlu zdrojového fyzického souboru (viz [dále](#)). Tam se v zadávacím okně ukáže stejná rozevírací nabídka obsahující stejný typ. Obě nabídky jsou synchronizovány.

Poznámka: Později může být tento typ změněn pouze CL příkazem WRKMBRPDM.

PC charset

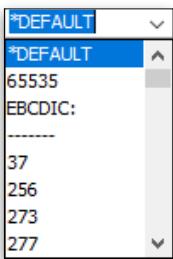


Parametr *PC charset* se uplatní při kopírování dat a při zobrazování v některých situacích. V seznamu je také speciální hodnota *DEFAULT. Interpretuje se podle kontextu. Podrobnosti jsou uvedeny v dalších kapitolách.

Uživatel může znakovou sadu (character set) zvolit z rozbalovacího seznamu nebo může její kód zapsat sám do vstupního pole.

Poznámka: Na rozdíl od IFS souborů nemají PC soubory žádný svůj atribut znakové sady. Proto se parametr *PC charset* při vytváření nového PC souboru neuplatní.

IBM i CCSID



Parametr *IBM i CCSID* se uplatní při kopírování dat a při zobrazování. Uplatní se také při vytváření nového zdrojového souboru (source physical file), při vytváření nového IFS souboru (stream file) nebo při zobrazování a kopírování tiskového souboru (spooled file). Podrobnosti jsou uvedeny v dalších kapitolách.

Uživatel může kód CCSID (Coded Character Set ID) zvolit z rozbalovacího seznamu nebo jej může zapsat sám do vstupního pole.

Source line length

Údaj určuje délku řádku zdrojového textu při vytváření nového *zdrojového souboru*. Například pro zdrojový soubor QDDSSRC je standardně volena délka 80, pro soubor QRPGLESRC délka 100 atd. Lze však zapsat jinou délku a potvrdit klávesou *Enter*.

Při kopírování textového PC souboru do zdrojového členu může být řádek (nebo jeho textová část) PC souboru delší než délka datové části zdrojového záznamu (definovaná při vytváření zdrojového souboru). Pak se řádek zkrátí tak, aby se vešel do datové části zdrojového záznamu.

Complete source record

Zaškrnuté políčko znamená při kopírování zdrojového členu do PC, že na začátek každého výstupního řádku se zařadí 12znakový údaj obsahující pořadové číslo a datum ze zdrojového záznamu.

Není-li políčko zaškrnuté, kopírují se do PC jen holá data zdrojového záznamu.

Obráceně, při kopírování textového PC souboru do zdrojového členu se toto políčko *neuplatní*. Aplikace se řídí podle prvního řádku PC souboru. Jsou-li první dva šestiznakové údaje celá čísla, dosadí se na začátek zdrojového záznamu.

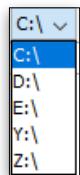
Nejsou-li to čísla, odvodí se chybějící údaje jako vypočtené pořadové číslo (od 1.00 po 1.00) a nulové datum. Tato čísla se předřadí řádku z PC a zapíší se společně jako záznam do zdrojového členu.

Tip: Volba kompletního zdrojového záznamu je užitečná jen tehdy, když potřebujeme zachovat pořadová čísla a data jednotlivých záznamů pro pozdější obnovu zdrojového členu ze zálohy v PC. V tomto případě je kopírovací operace dost pomalá.

Overwrite data

Zaškrnuté políčko znamená, že při kopírování je povoleno přepsat data existujícího souboru. Nezaškrnuté políčko nedovolí přepsat data existujícího souboru při kopírování ani při editaci.

Windows disks



Seznam disků je k dispozici jen v systému *Windows*. Ve skutečnosti jde o seznam kořenových adresářů souborového systému Windows. Uživatel vybere jeden ze seznamu. Předvolený je disk C:\.

Souborové systémy jako stromy

Souborový systém PC vlevo a souborový systém IBM i vpravo jsou prezentovány jako stromy. V prvním řádku stromu je umístěn kořen.

Při prvním spuštění aplikace je kořenem stromu kořen souborového systému.

- V systému Windows je to disk C:\.
- V systému typu unix je to dopředné lomítko / (tzv. root).
- V IBM i je to jako v systémech typu unix, tedy dopředné lomítko /.

Uživatel si může měnit kořen stromu pomocí rozbalovacího seznamu (combo box). Postupně, jak uživatel klepáním tlačítka myši rozevírá jednotlivé uzly stromu, se do *seznamu* zapisují cesty k příslušným objektům. Jestliže pak uživatel vybere ze seznamu některou cestu (zpravidla k adresáři), stane se vybraná cesta kořenem nového stromu a nový strom se zobrazí.

Uživatel může také zapsat cestu k objektu so vstupního pole rozbalovacího seznamu ručně a stisknout klávesu *Enter*, čímž nastaví nový kořen a zobrazí se nový strom.

Aplikace si aktuální kořen ukládá. Při příštím startu se zobrazí naposled uložený kořen a jemu odpovídající strom.

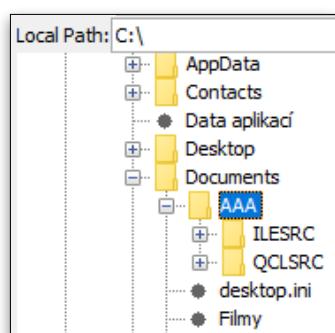
Rozevírání uzlů

Uzel stromu je třeba rozevřít *klepnutím* na jeho *text* nebo *ikonu* levým tlačítkem myši. Je to nutné, aby se uzly druhé úrovně naplnily příslušnou informací pro jejich další rozevření.

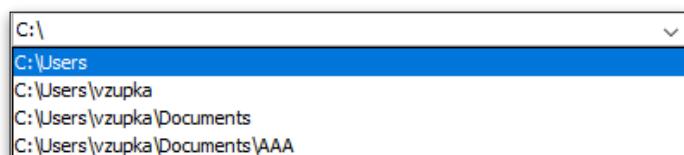
Upozornění: Nepoužívejte dvojité klepnutí na text nebo ikonu ani klepnutí na symbol pro rozevírání (znaménko + ve čtverečku v systému Windows nebo trojúhelníček v systému macOS)! To funguje pouze tehdy, byl-li již uzel předtím řádně rozevřen klepnutím na text nebo ikonu. Pak lze tímto způsobem uzel sevřít nebo znova rozevřít.

Levý strom – PC

Klepnutí *levým tlačítkem* myši na uzel (na obrázku AAA) zobrazí jeho objekty (zde adresáře ILESRC a QCLSRC).

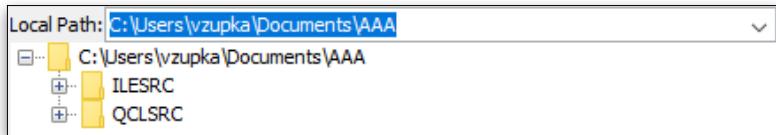


Zároveň se do seznamu zapíše cesta vedoucí k tomuto uzlu. Rozbalíme-li seznam "Local Path", uvidíme všechny cesty vedoucí k dosud rozevřeným uzlům:



Vybereme-li poslední adresář, ten se dosadí do vstupního pole a strom se zobrazí od nového kořene:

C:\Users\vzupka\Documents\AAA.



Pravý strom - IBM i

Způsob výběru objektů a kořenu je v pravém stromu stejný jako v levém. Použijeme rozbalovací seznam "Remote path" stejně jako "Local path".

Navíc lze vybírat knihovní objekty pomocí *vzorků* ve vstupních polích *LIB*, *FILE*, *MBR*. Výběr se spustí klepnutím na tlačítko *Connect/Reconnect* nebo stiskem klávesy *Enter* (když je ukazatel uvnitř některého ze jmenovaných polí).

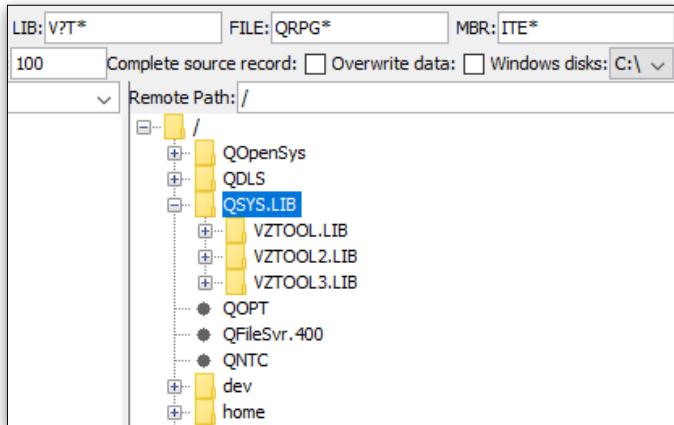
Vybíráme-li knihovní objekty (uvnitř uzlu /QSYS.LIB) pouze pomocí rozbalovacího seznamu "Remote path", dostaneme nový strom obsahující pouze vybrané knihovní objekty. Ostatní IFS objekty se ztratí.

Chceme-li vidět také ostatní IFS objekty, použijeme pro výběr vstupní pole *LIB*, *FILE*, *MBR*. Pak, jestliže rozevřeme uzel /QSYS.LIB, dostaneme nový strom obsahující vybrané knihovní objekty zároveň s ostatními IFS objekty. Tento způsob může být užitečný při kopírování mezi knihovními objekty a ostatními IFS objekty.

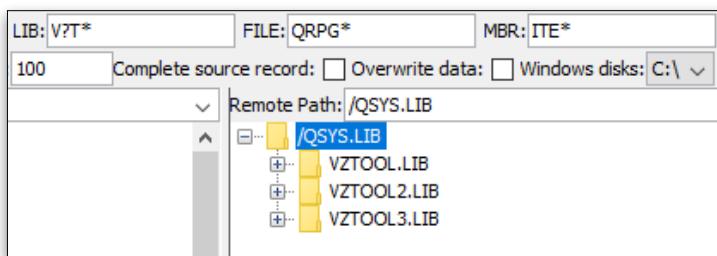
Obě metody lze kombinovat.

Výběr knihovních objektů

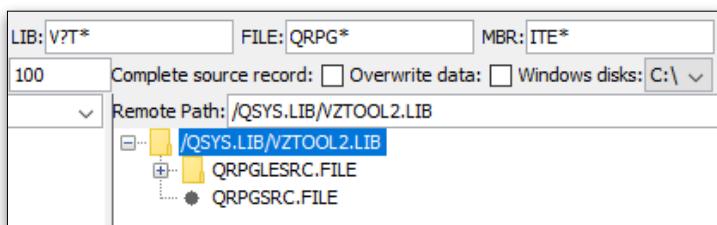
Vybereme-li knihovní objekty pouze pomocí polí LIB, FILE, MBR a ve vstupním poli rozbalovacího seznamu "Remote path" máme symbol / (kořen souborového systému IFS), dostaneme nový strom s knihovními objekty ukrytými v uzlu /QSYS.LIB. Abychom ty objekty viděli, musíme rozevřít tento a další uzly. Ostatní IFS objekty jsou stále vidět.



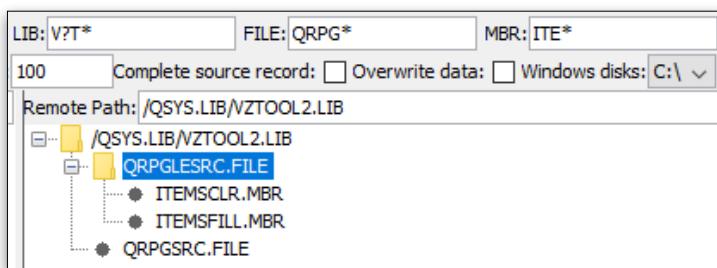
Zapíšeme-li do vstupního pole rozbalovacího seznamu "Remote path" cestu /QSYS.LIB a do vstupních polí LIB, FILE, MBR vzorky, dostaneme nový strom s kořenem /QSYS.LIB bez ostatních IFS objektů:



Zapíšeme-li do vstupního pole rozbalovacího seznamu "Remote path" cestu /QSYS.LIB/VZTOOL2.LIB a do vstupních polí LIB, FILE, MBR vzorky, omezíme strom na jedinou knihovnu VZTOOL2 s vybranými soubory



a vybranými členy:

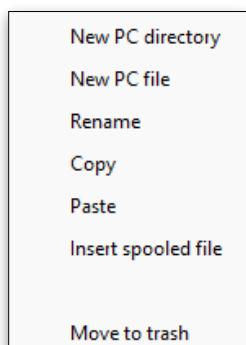


Kontextové nabídky pro PC

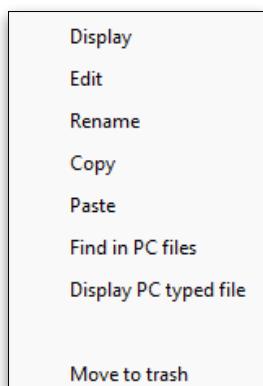
Klepnutí *pravým tlačítkem myši* na jeden nebo více vybraných uzelů vyvolá kontextovou nabídku s příkazy.

Na levé straně hlavního okna jsou k dispozici jen dvě nabídky.

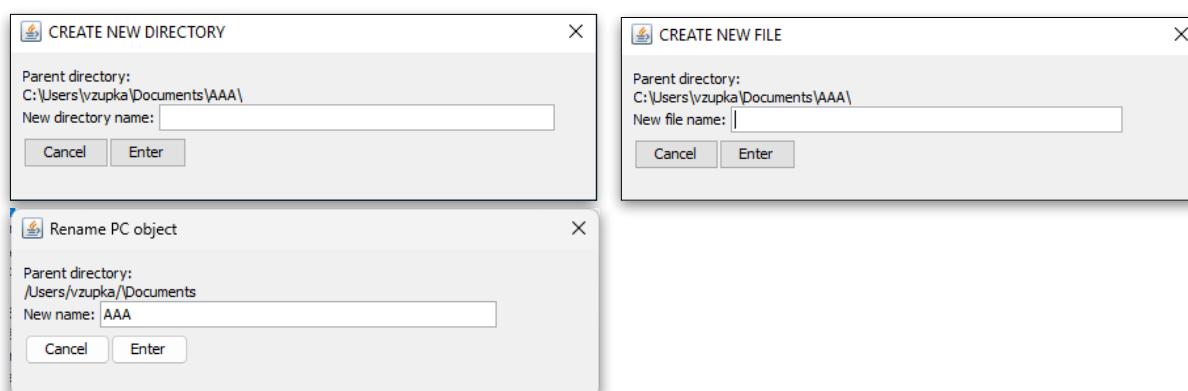
PC adresář



PC soubor



Příkazy *New PC directory*, *New PC file*, *Rename* vyvolají dialog jednotného tvaru. Adresáře a soubory v PC neobsahují žádnou informaci o znakové sadě.



Příkaz *Insert spooled file* je vysvětlen v kapitole [Tiskové soubory \(spooled files\)](#).

- Příkaz *Edit* vyvolá textový editor (viz kapitolu [Editace souboru](#)). Editor lze také vyvolat *poklepáním* (dvojitým klepnutím) na *uzel PC soubor*.
- Příkaz *Copy* si pamatuje soubory nebo adresáře z *vybraných uzelů* (výchozích uzelů).
- Příkaz *Paste* vloží nebo nahradí soubory nebo adresáře v *cílovém uzelu*.
- Příkaz *Find in PC files* na *vybraných souborech* vyvolá okno, kde lze zadat textový vzorek. V tom okně se zobrazí seznam souborů, v nichž se vzorek nalezl. Viz kapitolu [Hledání ve více souborech](#).
- Příkaz *Display PC typed file* zobrazí i netextový soubor, který má v operačním systému standardní typ (koncovku), například PDF, JPG, CSV, a další (viz kapitolu [Zobrazování ostatních souborů](#)).

Poznámka 1: Pozor při kopírování do několika cílových uzelů. Výchozí uzly se kopírují jen do *prvního* z cílových uzelů.

Poznámka 2: Příkazy *Copy* a *Paste* se použijí, když výchozí a cílový uzel *nejsou vidět zároveň*, a nelze tedy použít metodu *táhnout a pustit* (drag and drop).

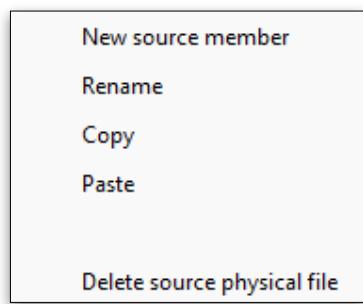
Kontextové nabídky pro IBM i

Na pravé straně hlavního okna je nabídek více. Některé příkazy nabídek mají podobnou funkci jako na levé straně.

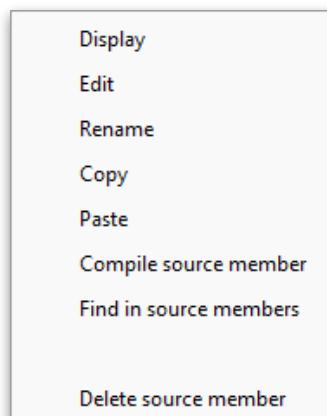
Nabídky pro IFS soubory (*IFS stream file*) a pro zdrojové členy (*Source member*) mají příkazy navíc:

- *Change CCSID* (viz kapitolu [Změna kódování v IFS](#))
- *Compile IFS file* a *Compile source member* (viz kapitolu [Kompilace](#))
- *Find in source members* a *Find in IFS files*. Viz kapitolu [Hledání ve více souborech](#).

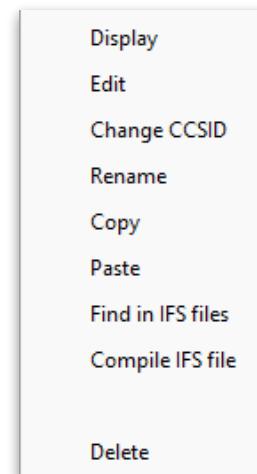
Source physical file



Source member

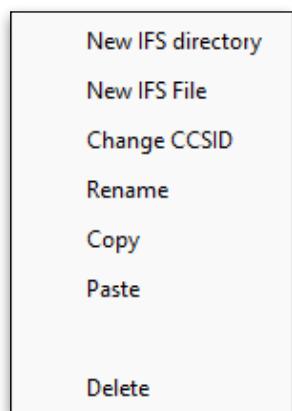


IFS stream file

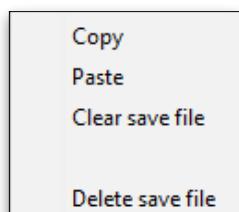


Ostatní příkazy jsou celkem zřejmé.

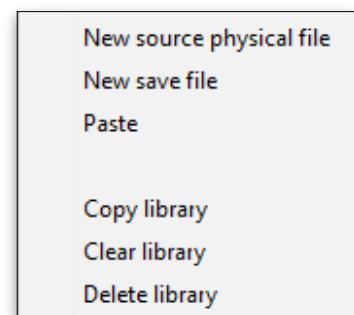
IFS directory



Save file



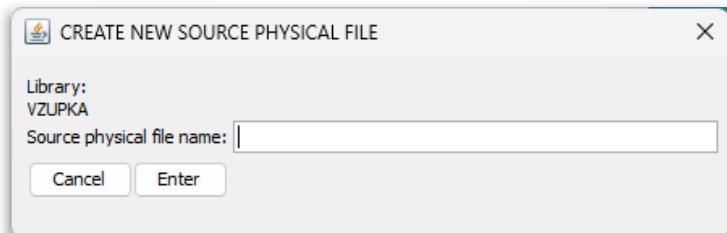
Library



Vytváření adresářů a souborů v IBM i

V pravé straně hlavního okna na zvoleném uzlu zvolíme z [kontextové nabídky](#) příkaz *New...*

Zdrojový fyzický soubor

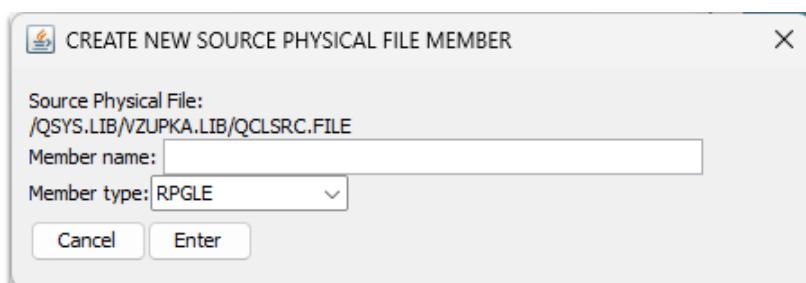


Nový *zdrojový soubor* získá CCSID z parametru *IBM i CCSID*.

- Je-li jeho hodnota ***DEFAULT**, je nahrazena hodnotou 500 (**EBCDIC ISO-8859-1**).
- Délka dat ve zdrojových záznamech se získává z parametru *Source line length*. Celý záznam je o 12 bajtů delší. Před daty je v prvních 6 bajtech pořadové číslo a v dalších 6 bajtech datum tvaru YYDDMM.

Zdrojový člen

Nový *zdrojový člen* získá typ z rozevírací nabídky *Member type*, který je také zobrazen



v rozevírací nabídce *Source type* hlavního okna. Obě nabídky se při změně kterékoliv z nich mění současně.

Poznámka: Později může být tento typ změněn pouze CL příkazem WRKMBRPDM.

Nový *zdrojový člen* získá CCSID podle zdrojového souboru v němž se vytváří, jako svůj atribut, který si stále zachovává. Délka záznamů se získává z atributu "Maximum record length" zdrojového fyzického souboru.

IFS adresář

Nový *IFS adresář* získá CCSID z parametru *IBM i CCSID* jako svůj atribut. Je-li jeho hodnota ***DEFAULT**, je nahrazena hodnotou 819 (**ASCII ISO-8859-1**).

IFS soubor

Nový *IFS soubor* získá CCSID z parametru *IBM i CCSID* jako svůj atribut. Je-li jeho hodnota ***DEFAULT**, je nahrazena hodnotou 819 (**ASCII ISO-8859-1**).

Poznámka: V IFS souboru i v IFS adresáři lze hodnotu CCSID dodatečně změnit (viz [zde](#)).

Způsoby kopírování

V aplikaci lze kopírovat soubory i adresáře, a to všemi směry:

- IBM i ↔ PC
- PC ↔ IBM i
- IBM i ↔ IBM i
- PC ↔ PC

Kopírování lze provádět dvěma způsoby. První způsob používá příkazy *Copy* a *Paste* z kontextových nabídek. Druhý způsob je přetažení myší a upuštění (*drag and drop*).

Kopírování z IBM i do PC

- zdrojový člen → PC soubor
- zdrojové členy → PC adresář
- zdrojové soubory → PC adresář
- IFS soubor → PC soubor
- IFS soubory → PC adresář
- IFS adresář → PC adresář
- ukládací soubor (save file) → PC soubor
- ukládací soubory (save files) → PC adresář

Zdrojový člen → PC soubor

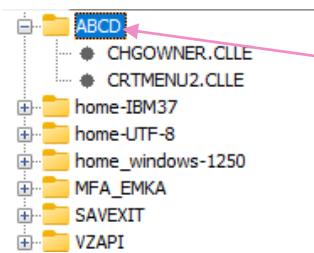
Při kopírování *zdrojového členu* do PC souboru se data zdrojového členu nejprve překódují podle parametru *IBM i CCSID* do *pomocného textu* kódovaného UTF-16 a pak se překódují podle parametru *PC charset*.

- Je-li v parametru *IBM i CCSID* zadána hodnota **DEFAULT*, data se do pomocného textu překódují podle *CCSID atributu* zdrojového členu.
- Je-li v parametru *PC charset* zadána hodnota **DEFAULT*, data z pomocného textu se překódují podle znakové sady *ISO-8859-1* (Latin-1).

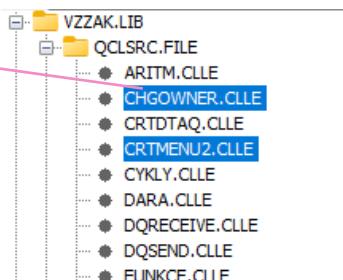
Poznámka: Zdrojový soubor může být vytvořen s atributem CCSID 1208. Zdrojový člen vytvořený v tomto zdrojovém souboru má pak stejný atribut. Potom, jestliže parametr IBM i CCSID nebo CCSID atribut je 1208 (ekvivalent UTF-8), nahradí se při překódování do pomocného textu hodnotou 65535. To proto, aby se do UTF-16 nepřekódovaly jednotlivé bajty, ale celé znaky UTF-8.

Zdrojové členy → PC adresář přetažením

PC

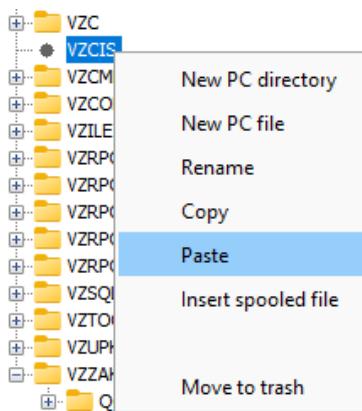


IBM i

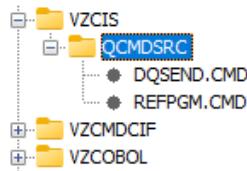
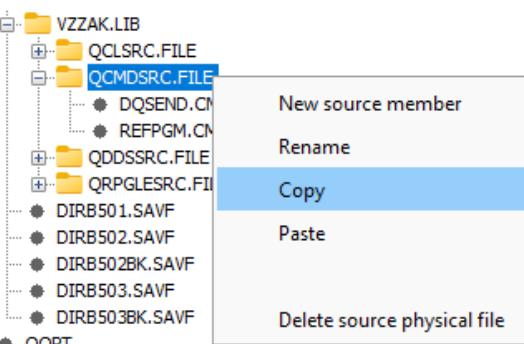


Zdrojový soubor → PC adresář copy-paste

PC



IBM i



IFS soubor (stream file) → PC soubor

Při kopírování *IFS souboru* (stream file) do PC souboru se data překódují podle parametrů *IBM i CCSID* a *PC charset*.

Je-li v obou parametrech zadána hodnota **DEFAULT*, přenáší se data *bez změny (binárně)*.

Jinak se data překódují podle parametru *IBM i CCSID* do pomocného textu kódovaného UTF-16 a pak se překódují podle parametru *PC charset*.

- Je-li hodnota **DEFAULT* zadána jen v parametru *IBM i CCSID*, data se do pomocného textu překódují podle CCSID atributu IFS souboru.
- Je-li hodnota **DEFAULT* zadána jen v parametru *PC charset*, data z pomocného textu se do PC souboru překódují podle znakové sady *ISO-8859-1* (Latin-1).

Ukládací soubor (save file) → PC soubor

Ukládací soubor je uložen v knihovně jako objekt typu **FILE* s atributem *SAVF*.

Kopírováním do PC adresáře dostane nový soubor příponu *.savf*. Existující PC soubor musí mít příponu *.savf*, aby se do něj dal ukládací soubor kopírovat.

Kopírování z PC do IBM i

- PC soubor → zdrojový člen
- PC soubor → zdrojový soubor
- PC adresář → zdrojový soubor
- PC soubor → IFS soubor
- PC soubory → IFS adresář
- PC adresáře → IFS adresář
- PC soubor → ukládací soubor (save file)

PC soubor → zdrojový člen

Při kopírování z PC souboru *do zdrojového členu* se data PC souboru překódují ze sady zadанé v parametru *PC charset* do sady zadáné *CCSID atributem* zdrojového členu. Parametr *IBM i CCSID* se přitom *ignoruje*. Výsledné znaky nemusí být v souladu s cílovou sadou.

Poznámka 1: Jestliže je hlášena chyba při kopírování souborů, může se vytvořit prázdný soubor.

Poznámka 2: Přenos může trvat docela dlouho, jestliže se znakové sady liší a je-li zdrojový člen dlouhý. Program musí číst vstupní soubor po řádcích a každý řádek před zápisem překódovat.

PC soubor → IFS soubor

Při kopírování z PC souboru *do IFS souboru* se data PC souboru překódují ze znakové sady zadané v parametru *PC charset* do sady zadáné v parametru *IBM i CCSID*.

Je-li v obou parametrech zadána hodnota **DEFAULT*, přenášeji se data *beze změny (binárně)*.

Jinak se data překódují podle parametru *PC charset* interně do pomocného textu kódovaného UTF-16, který se pak překóduje podle parametru *IBM i CCSID*.

- Je-li v parametru *PC charset* zadána hodnota **DEFAULT*, dosadí se místo ní hodnota *ISO-8859-1* (ASCII ISO-8859-1, Latin Alphabet No. 1).
- Je-li v parametru *IBM i CCSID* zadána hodnota **DEFAULT*, dosadí se místo ní hodnota *819* (ASCII ISO-8859-1, Latin Alphabet No. 1).
- Jestliže si znakové sady na obou stranách odpovídají, přenesou se data v některých případech bez překódování. Jde o tyto případy:
 - ISO-8859-1 → CCSID 819, CCSID 858
 - windows-1250, Cp1250 → CCSID 1250
 - windows-1251, Cp1251 → CCSID 1251
 - UTF-8 → CCSID 1208
 - UTF-16 → CCSID 1200, CCSID 13488
 - IBM500, Cp500 → CCSID 500
 - IBM870, Cp870 → CCSID 870

PC soubor → ukládací soubor (save file)

PC soubor označený příponou *.savf* se považuje za ukládací soubor a kopíruje se jako objekt typu **FILE* s atributem SAVF do knihovny. Tentýž soubor lze kopírovat i do IFS souboru nebo adresáře beze změny přípony aniž ztrácí svůj obsah jako ukládací soubor.

Kopírování z IBM i do IBM i

- zdrojový člen → zdrojový člen
- zdrojové členy → zdrojový soubor
- zdrojové soubory → knihovna
- zdrojový člen → IFS soubor
- zdrojové členy → IFS adresář
- zdrojové soubory → IFS adresář
- FS soubor → IFS soubor
- IFS soubory → IFS adresář
- IFS adresáře → IFS adresář
- IFS adresáře → zdrojový soubor
- ukládací soubory → IFS adresář
- ukládací soubor → IFS soubor
- ukládací soubory → IFS adresář
- ukládací soubory v IFS → knihovna
- knihovna → nová knihovna

Poznámka: Kopírování IFS adresářů a souborů (kromě ukládacích souborů) do knihovny je zakázáno.

Zdrojový člen → zdrojový člen

Při kopírování zdrojového členu do zdrojového členu (existujícího nebo nového) se data překódují podle zdrojových souborů. v nichž leží. Tedy z CCSID atributu výchozího zdrojového souboru do CCSID atributu cílového zdrojového souboru.

Parametr *IBM i CCSID se ignoruje*.

Zdrojový člen → IFS soubor

Při kopírování zdrojového členu do existujícího IFS souboru se data překódují do CCSID atributu IFS souboru.

Při kopírování zdrojového členu do dosud neexistujícího IFS souboru, který se právě vytváří, se data nepřekódují a nový IFS soubor převezme CCSID atribut ze zdrojového souboru, v němž zdrojový člen leží.

V obou případech se parametr *IBM i CCSID ignoruje*.

IFS soubor → zdrojový člen

Při kopírování IFS souboru do zdrojového členu (existujícího nebo nového) se data překódují z CCSID atributu IFS souboru do CCSID atributu zdrojového souboru bez ohledu na to, zda zdrojový člen již existuje nebo se právě vytváří.

Parametr *IBM i CCSID se ignoruje*.

IFS soubor → IFS soubor

Má-li parametr *IBM i CCSID* hodnotu *DEFAULT nebo oba soubory mají stejně CCSID atributy, data se přenášejí bez změny (binárně).

Jinak se při kopírování IFS souboru do IFS souboru data překódují z CCSID atributu vstupního souboru do CCSID atributu výstupního souboru.

Při kopírování do dosud *neexistujícího* IFS souboru, který se *právě vytváří*, nový IFS soubor převezme CCSID atribut ze vstupního IFS souboru a data se tudíž přenášejí *beze změny* (*binárně*).

Ukládací soubor v knihovně → IFS

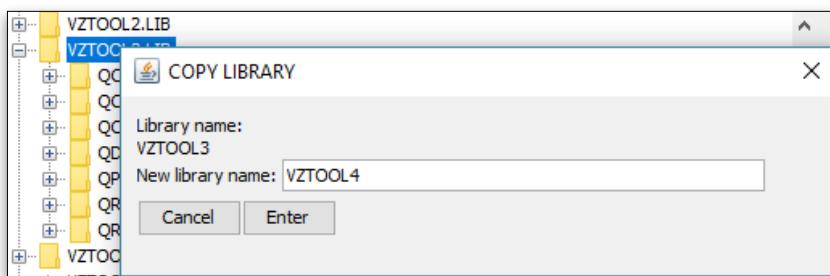
Ukládací soubor v knihovně je objekt typu *FILE s atributem SAVF. Jestliže výstupní IFS soubor existuje, musí mít příponu .savf. Nově vytvořený IFS soubor získá příponu .savf.

Ukládací soubor v IFS adresáři → knihovna

IFS soubor s příponou .savf se považuje za ukládací soubor (save file). Jestliže ukládací soubor v knihovně existuje, má příponu .SAVF. Nově vytvořený ukládací soubor získá příponu .SAVF.

Knihovna → knihovna

Knihovnu lze zkopirovat pod jiným jménem, které zadá uživatel.



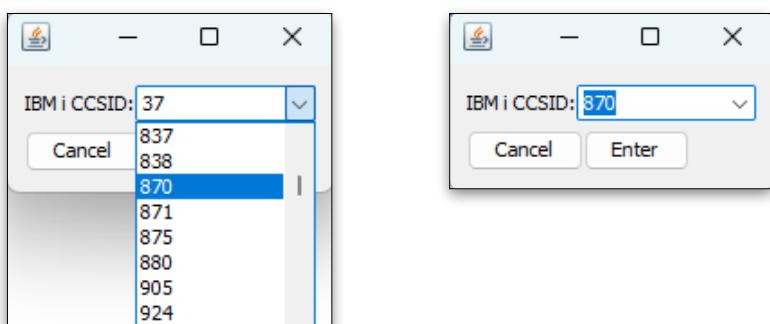
Kopírování z PC do PC

- PC soubor → PC soubor
- PC soubory → PC adresář
- PC adresář → PC adresář

Při kopírování se neprovádějí žádné konverze dat.

Změna kódování v IFS

Z kontextové nabídky se zobrazí okno s rozbalovacím seznamem. Uživatel může kód CCSID (Coded Character Set ID) zvolit z rozbalovacího seznamu nebo jej může zapsat sám do vstupního pole. Stiskem *Enter* provede změnu atributu CCSID v IFS objektu. Je-li objektem adresář, změní se atribut ve všech podřízených objektech.



Zobrazování textových souborů

Textové soubory v IBM i a v PC lze zobrazit výběrem příkazu *Display* z kontextové nabídky.

Obsah souboru se zobrazí v samostatném okně s informací o *znakové sadě* textu v souboru.

```
Find: outputParam 01 / 03 ◀ ▶ Aa Monospaced 12 CCSID 870 was used for display.  
000100170807**free  
000200170807  
000300170807// Main program called from Java or from RPG with parameters  
000400170807dcl-ds parameter3 template;  
000500170807    outputParam char(30);           // returned value  
000600170807    quantity int(5) overlay(outputParam); // input value  
000700170807end-ds;  
000800170807  
000900170807dcl-pi *N; // Program interface  
001000170807    nametext char(8);      // input  
001100170807    answer char(50);       // output  
001200170807    outPar likeds(parameter3); // in-out  
001300170807end-pi;  
001400170807  
001500170807dcl-s padding char(100) inz(*all'-');  
001600170807  
001700170807answer = 'ANSWER(50): ' + %char(outPar.quantity) + padding; // output  
001800170807outPar.outputParam =  
001900170807    'NAME+QUANTITY: ' + nametext + '+' + %char(outPar.quantity) + padding; // output  
002000170807  
002100170807dump(a) 'TESTPROG';  
002200170807  
002300170807return;
```

V horní části okna jsou ovládací komponenty:

- *Find* – zapisování textového vzorku do pole vyhledává odpovídající texty. Čísla v poli ukazují pořadové číslo shody a počet shod.
- ◀ ▶ *Tlačítka s šipkami* – vyhledávají předchozí a následující shodný vzorek v souboru. Po klepnutí šipka zčerná a ukazuje směr hledání. Opačná šipka zešedne. Stejnou funkci mají klávesové zkratky Ctrl ↑ (předchozí) a Ctrl ↓ (další).
- **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.
- *Velikost písma* – vstupní pole určuje velikost písma v textu zobrazeného souboru.

Klávesa ESC ukončí zobrazení a odstraní okno.

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

Hledání ve více souborech

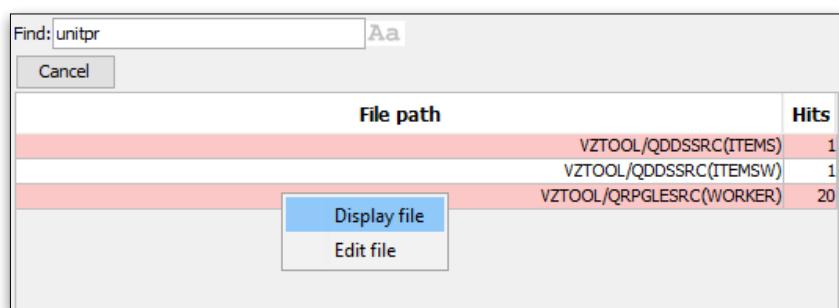
Vybereme jeden nebo více souborů (PC souborů, zdrojových členů nebo IFS souborů), které mohou a nemusí sousedit. Příkaz *Find in . . .* z kontextové nabídky vyvolá okno, v němž zadáme textový vzorek a stiskneme klávesu Enter.

Je-li ikona "Match case" (Aa) světle šedá, hledá se bez ohledu na velikost písmen. Je-li černá (Aa), hledá se totožný text.

Jestliže se vzorek našel v jednom nebo více souborech, objeví se jejich seznam v okně.

Uživatel může vybrat jeden nebo několik z nich a bud' je zobrazit nebo editovat. Klepnutí pravým tlačítkem vyvolá kontextovou nabídku s příkazy *Display file* a *Edit file*. Tyto příkazy otevřou okno se zobrazeným souborem, nebo okno editoru. V obou případech jsou všechny výskyty nalezeného vzorku zvýrazněny v závislosti na barvě ikony Match case.

Například uživatel vybral několik (nesousedících) zdrojových členů domněle obsahujících vzorek *unitpr* a zvolil na jednom z nich příkaz *Find in source members*. Pak zapsal vzorek do pole *Find* a stiskl klávesu Enter. Vzorek se našel ve třech zdrojových členech.



Uživatel vybral dva řádky a zvolil příkaz *Display file*. Ve dvou oknech se zobrazí se dva členy, display file ITEMS a program WORKER se všemi zvýrazněnými výskyty nalezeného textu.

```
Display member '/QSYS.LIB/VZTOOL.LIB/QRPGLESRC.FILE/WORKER.MBR'
Find: unitpr          01 / 20  ◀ ▶ Aa Monospaced 12 CCSID 870 was used for display.

006700161119
006800161119
006900161119      *=====
007000161119      *   Data definitions
007100161119      *=====
007200161119
007300161119      *   Socket data structure for input from the client
007400161119      D SocketData    DS      79
007500161119      D Record           Overlay(SocketData)
007600161119      D msgID            2A
007700161119      D                   Overlay(Record)
007800161119      *   Database fields (numerics edited)
007900161119      D AITEMNBR        Like(IITEMNBR)
008000161119      D                   Overlay(Record: *Next)
008100161119      D AUNITPR         11A           edited UNITPR
008200161119      D                   Overlay(Record: *Next)
008300161119      D AAMOUNT          11A           edited AMOUNT
```

Zobrazování ostatních (netextových souborů)

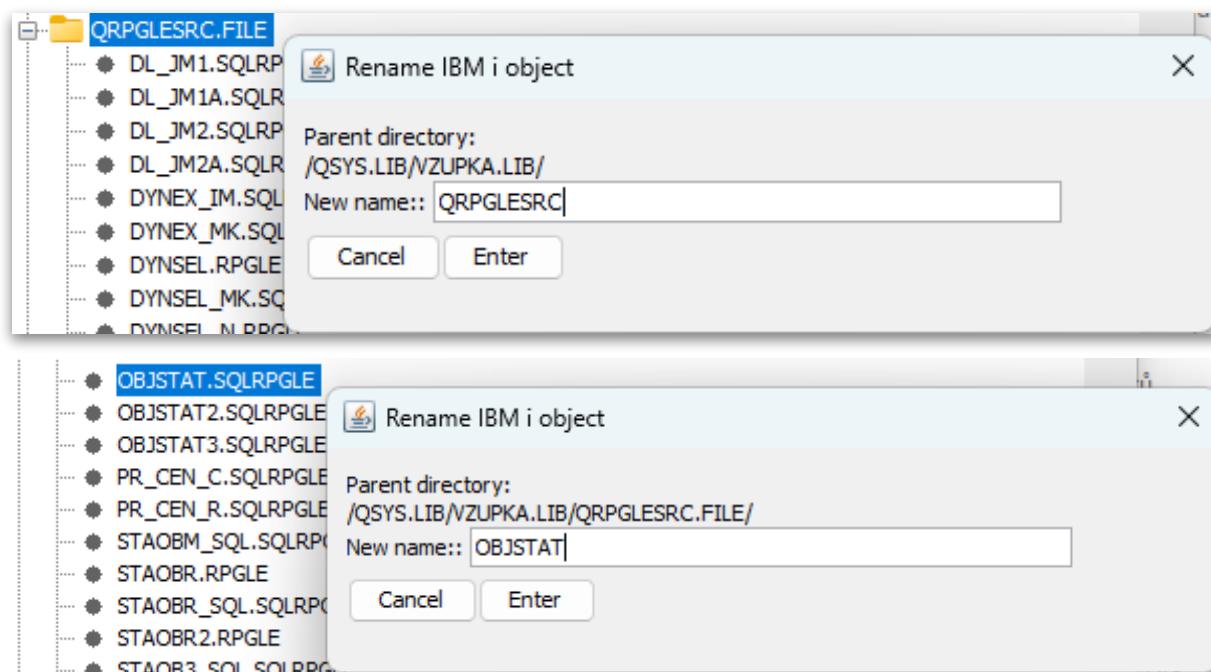
Soubory v PC označené známou koncovkou, například PDF, JPG, CSV a mnohé další, lze zobrazit výběrem příkazu *Display PC typed file* z kontextové nabídky, jestliže jsou jim v operačním systému určené aplikace.

Poznámka: V systému *macOS* se nezobrazují soubory, k nimž vede nesprávně kódovaná cesta (název adresáře nebo souboru obsahuje jiné znaky než ASCII nebo mezery).

Přejmenování objektů

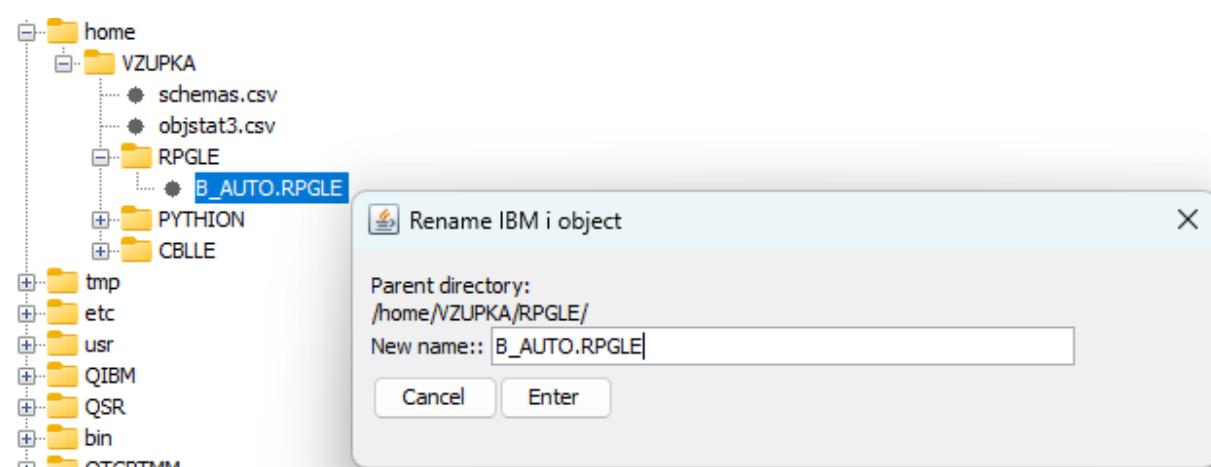
Volbou příkazu *Rename* v kontextové nabídce se zobrazí výzva k přepsání jména objektu.

U knihovních objektů (/QSYS.LIB) nelze měnit typ, jen holé jméno.



U IFS objektů (neknihovních) lze měnit celý text jména.

U PC objektů lze měnit celé jméno.



Zobrazování a editace v PC – znakové sady

Soubory se zobrazují a editují s použitím parametru *PC charset*. Je-li v něm zadána hodnota *DEFAULT, dosadí se *ISO-8859-1*. Jestliže soubor obsahuje neplatné znaky, hlásí se chybová zpráva. Uživatel může změnit parametr a učinit nový pokus.

Zobrazování a editace v IBM i – znakové sady

Zdrojové členy se zobrazují a editují s použitím svého *CCSID atributu* bez ohledu na hodnotu parametru *IBM i CCSID*. Znaky se zobrazí nesprávně, jestliže danému CCSID atributu nevyhovují.

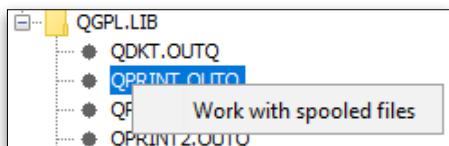
IFS soubory se zobrazí i editují s použitím svého *CCSID atributu* bez ohledu na parametr *IBM i CCSID*. Znaky se zobrazí nesprávně, jestliže danému CCSID atributu nevyhovují.

Tiskové soubory se zobrazují (ale needitují) s použitím parametru *IBM i CCSID*. Zobrazují-li se nesprávné znaky, může pomoci použití hodnoty *DEFAULT nebo 65535. Program se pak snaží odhadnout správné zakódování.

Tiskové soubory (spooled files)

Tiskové soubory můžeme zobrazit a také kopírovat do PC. Můžeme postupovat dvěma způsoby.

První způsob vyžaduje znalost, které knihovny obsahují objekty výstupní tiskové fronty (typ objektu *OUTQ), např. knihovny QGPL nebo QUSRSYS. Pravým tlačítkem myši klepneme na uzel s koncovkou .OUTQ, čímž zobrazíme nabídku jednoho příkazu *Work with spooled files*.



Ve druhém způsobu nemusíme znát jméno knihovny. Místo toho použijeme systémovoou knihovnu QSYS.LIB. Ta neobsahuje výstupní fronty, ale slouží jako místo pro zobrazení nabídky s příkazem *Work with spooled files*. Tak můžeme pracovat se *všemi* tiskovými soubory.

Příkaz *Work with spooled files* vyvolá okno s tabulkou tiskových souborů.

File name	File num.	Pages	Job name	User	Job num.	Date	Time	
				QSYS				
				VZUPKA				
File name	File num.	Pages	Job name		Job num.	Date	Time	Output queue
QPRINT	1	1	QLSPSVR	QSYS	032508	1170331	134516	QGPL/QPRINT
CENPOR	6	10	QPRTJOB	VZUPKA	033626	1170404	183428	QGPL/QPRINT
TESTPROG	7	7	QPRTJOB	VZUPKA	033626	1170405	102552	QGPL/QPRINT

Práci s tiskovými soubory můžeme ukončit uzávěrem okna a také *klávesou ESC*.

K omezení rozsahu tabulky můžeme zadat podmínky do vstupních polí nad tabulkou a stisknout klávesu Enter. Text zapsaný v poli se hledá v odpovídajícím sloupci tabulky. Podmínky zadané ve více polích se vyhodnocují současně. Prázdné vstupní pole neomezuje tabulku.

Pole *User* je rozbalovací seznam jmen všech uživatelů (profilů), jimž zobrazené tiskové soubory patří. Výběrem jednoho jména omezíme tabulku na tiskové soubory patřící tomu uživateli.

Poklepání (dvojité klepnutí - dvojklik) na řádek zobrazí tiskový soubor *přímo* v samostatném okně (viz [Zobrazování souborů](#) výše).

Chceme-li dále pracovat s tiskovými soubory, ivybereme jeden řádek nebo skupinu řádků levým tlačítkem myši. Při výběru souvislé skupiny podržíme klávesu *Shift*, k výběru nesouvislé skupiny podržíme klávesu *Ctrl* (v macOs *Cmd*).

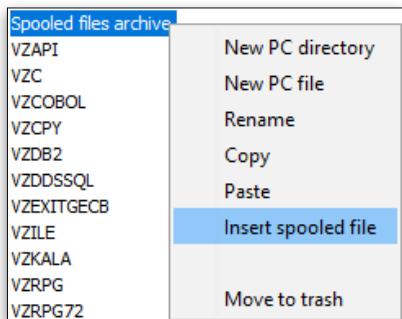
Pak *pravým* tlačítkem myši zobrazíme nabídku příkazů:

QPJOBLOG	104	4	QPRTJOB	QSYS	1170415	1170414	000023	QUSRSYS/QEZJOBLOG
QPJOBLOG	105	7	QPRTJOB	QSYS		1170414	000025	QUSRSYS/QEZJOBLOG
QPJOBLOG	20	14	QPRTJOB	VZUPKA		1170414	094333	QUSRSYS/QEZJOBLOG
QPJOBLOG	106	4	QPRTJOB	QSYS		1170415	000031	QUSRSYS/QEZJOBLOG
QPJOBLOG	107	7	QPRTJOB	QSYS		1170415	000033	QUSRSYS/QEZJOBLOG
VEKDS04	21	8	QPRTJOB	VZUPKA	033626	1170415	072849	QGPL/QPRINT

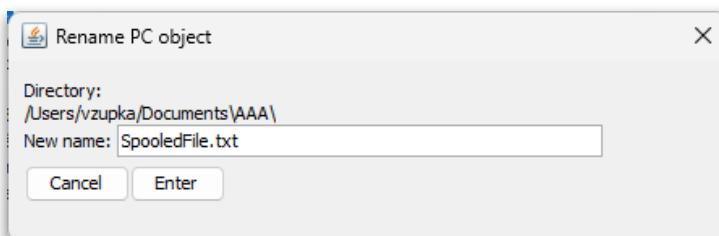
<i>Display</i>	zobrazit vybrané tiskové soubory
<i>Copy</i>	kopírovat tiskový soubor (jeden nebo poslední z vybraných)
<i>Delete</i>	zrušit vybrané tiskové soubory

Příkazy *Display* a *Copy* převedou tiskový soubor z binárního tvaru do textu, který se zapíše do interního textového souboru *SpooledFile.txt* v adresáři *workfiles*. Příkaz *Display* pak ještě zobrazí text tiskového souboru v samostatném okně (viz [Zobrazování souborů](#) výše).

Interní soubor můžeme ještě zkopirovat pod stejným nebo jiným jménem do zvoleného adresáře v PC, abychom s ním mohli dále manipulovat (např. vytisknout nebo poslat poštou). Z nabídky u zvoleného adresáře levého stromu zvolíme příkaz *Insert spooled file*.



Příkaz vyvolá dialog s výzvou k určení jména souboru.



Po změně nebo ponechání jména v dialogu se soubor zapíše do zvoleného adresáře.