

SOURCE FILE CDJSON/QRPGLESRC

MEMBER JSONINV01

```

SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8 ...+... 9 ...+... 0
100 **FREE
100 Ctl-Opt BndDir('JSONXML') dftactgrp(*NO) ACTGRP('QILE');
100 DCL-F lffakinv DISK(*EXT) USAGE(*INPUT) KEYED;
100 DCL-F lfstafaf DISK(*EXT) USAGE(*INPUT) KEYED;
100 DCL-F lfbew2 DISK(*EXT) USAGE(*INPUT) KEYED;
100 DCL-F pfsigi DISK(*EXT) USAGE(*INPUT) KEYED;
100 DCL-F lforderb DISK(*EXT) USAGE(*INPUT) KEYED;
100 DCL-F LFPROJFFA1 DISK(*EXT) USAGE(*INPUT) KEYED;
100 DCL-F lfvprs DISK(*EXT) USAGE(*INPUT) KEYED;
100 DCL-F lfkln DISK(*EXT) USAGE(*INPUT) KEYED;
100 DCL-F lfvern DISK(*EXT) USAGE(*INPUT) KEYED;
100 /include qrpgleRef,jsonxml
100
100 dcl-pi *n ;
100 pfirma char(2) ;
100 pdatum char(8) ;
100 end-pi ;
100
100 dcl-s pklnr char(1) ;
100
100 Dcl-S klnr zoned(8:0);
100 Dcl-S kbase zoned(9:2);
100 Dcl-S teller int(10);
100 Dcl-S pTemp Pointer;
100 Dcl-S pFakn Pointer;
100 Dcl-S pJson Pointer;
100 Dcl-S pNewObj Pointer;
100
100 pklnr = 'K';
100
100 kimpda = %dec(pdatum:8:0);
100
100 pNewObj = json_NewObject();
100
100 setll (kimpda) lffakinv;
100 reade (kimpda) lffakinv;
100 dow not %eof(lffakinv);
100 teller = teller + 1;
100 pJson = json_NewObject();
100 chain ('001':KFAKNR) lfbew2;
100 chain (pklnr:bnum) pfsigi;
100 json_SetStr (pJson : 'companyId' : pfirma );
100 json_SetInt (pJson : 'invoiceNumber' : kfaknr );
100 json_SetStr (pJson : 'voucherNumber' : bjnl + '/' + %char(bdocnr) );
100 json_SetStr (pJson : 'type' : %trim(baard) );
100 klnr = %dec(bnum:8:0);
100 monitor;
100 kklnr = klnr;
100 on-error;
100 kklnr = 0;
100 endmon;
100 json_SetStr (pJson : 'customerId' : pfirma + bnum);
100 json_SetNum (pJson : 'cashDiscount' : kkortb );

```

SOURCE FILE CDJSON/QRPGLESRC

MEMBER JSONINV01

```
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8 ...+... 9 ...+... 0
5400  json_SetStr      (pJson : 'currency' : 'EUR' );
5500  json_SetNum      (pJson : 'totalInvoiceAmount' : kbdrf );
5600  if bvald = *blanks;
5700  eval bvald = bimpd;
5800  endif;
5900  json_SetStr      (pJson : 'invoiceDate' : %subst(bimpd:1:4) + '-' +
6000  %subst(bimpd:5:2) + '-' + %subst(bimpd:7:2));
6100  json_SetStr      (pJson : 'dueDate' : %subst(bvald:1:4) + '-' +
6200  %subst(bvald:5:2) + '-' + %subst(bvald:7:2));
6300  json_SetNum      (pJson : 'totalVatAmount' : kbtw );
6400  kbase = kbdrf - kbtw - kkortb;
6500  json_SetNum      (pJson : 'totalBaseAmount' : kbase );
6600  json_SetStr      (pJson : 'vatCode' : bbtwc );
6700  json_SetStr      (pJson : 'description' : boms );
6800  json_SetStr      (pJson : 'invoiceCountry' : %trimr(LANDK));
6900  json_SetStr      (pJson : 'deliveryCountry' : %trimr(LANDK));
7000  json_SetInt      (pJson : 'orderNumber' : bordnr);
7100  chain bordnr lforderb;
7200  if %found;
7300  json_SetStr      (pJson : 'customerReference' : %trimr(OREFK));
7400  json_SetInt      (pJson : 'articleId' : oartnr);
7500  json_SetStr      (pJson : 'articleDescription' : %trimr(OARTUI));
7600  json_SetInt      (pJson : 'quantity' : ostuks);
7700  json_SetNum      (pJson : 'articlePrice' : ofaktw);
7800  json_SetNum      (pJson : 'installationPrice' : owerpl);
7900  else;
8000  json_SetStr      (pJson : 'customerReference' : '');
8100  json_SetInt      (pJson : 'articleId' : 0);
8200  json_SetStr      (pJson : 'articleDescription' : '');
8300  json_SetInt      (pJson : 'quantity' : 0);
8400  json_SetNum      (pJson : 'articlePrice' : 0);
8500  json_SetNum      (pJson : 'installationPrice' : 0);
8600  endif;
8700
8800  // pffakrek velden
8900  pfakn = json_newObject();
9000  json_SetNum      (pfakn : 'KEGAL' : KEGAL);
9100  json_SetNum      (pfakn : 'KBDR0' : KBDR0);
9200  json_SetNum      (pfakn : 'KBDR6' : KBDR6);
9300  json_SetNum      (pfakn : 'KBDR17' : KBDR17);
9400  json_SetNum      (pfakn : 'KBDR19' : KBDR19);
9500  json_SetNum      (pfakn : 'KBDR25' : KBDR25);
9600  json_SetNum      (pfakn : 'KBDR33' : KBDR33);
9700  json_SetNum      (pfakn : 'KBDR0M' : KBDR0M);
9800  json_SetNum      (pfakn : 'KKORTB' : KKORTB);
9900  json_SetNum      (pfakn : 'KBD195' : KBD195);
10000 json_SetNum      (pfakn : 'KBDR12' : KBDR12);
10100 json_SetStr      (pfakn : 'KKODE' : KKODE );
10200 json_SetStr      (pfakn : 'KMNP' : KMNP );
10300 json_MoveObjectInto(pjson : 'pffakrek' : pfakn);
10400 // einde pffakrek velden
10500
10600
```

SOURCE FILE CDJSON/QRPGLESRC
MEMBER JSONINV01

```
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8 ...+... 9 ...+... 0
10700  if pfirma = 'WV';                                28/04/23
10800  chain (bdocnr) LFPROJFFA1;                          16/05/23
10900  if %found;                                          28/04/23
11000  json_SetInt      (pJson : 'projectId' : FPROJN);    28/04/23
11100  else;                                                 28/04/23
11200  json_SetInt      (pJson : 'projectId' : 0);          28/04/23
11300  endif;                                                28/04/23
11400  else;                                                 28/04/23
11500  json_SetInt      (pJson : 'projectId' : 0);          28/04/23
11600  endif;                                                28/04/23
11700
11800  if pfirma = 'WV';                                28/04/23
11900  prnr = kklmr;                                         28/04/23
12000  chain (prnr) lfvprs;                                  28/04/23
12100  if %found;                                          28/04/23
12200  json_SetStr      (pJson : 'CustomerEnterpriseNumber' : PROND); 28/04/23
12300  else;                                                 28/04/23
12400  json_SetStr      (pJson : 'CustomerEnterpriseNumber' : '0'); 28/04/23
12500  endif;                                                28/04/23
12600  else;                                                 28/04/23
12700  json_SetStr      (pJson : 'CustomerEnterpriseNumber' : '0'); 28/04/23
12800  endif;                                                28/04/23
12900
13000  chain (kklmr) lfkln;                                  28/04/23
13100  if %found;                                          28/04/23
13200  chain (kvtmr) lfvtrn;                                02/05/23
13300  if %found;                                          28/04/23
13400  select;                                              02/05/23
13500  when vbank = 1;                                       02/05/23
13600  json_SetStr      (pJson : 'BusinessUnitDimemsion' : 'B2C'); 02/05/23
13700  when vbank = 2;                                       02/05/23
13800  json_SetStr      (pJson : 'BusinessUnitDimemsion' : 'B2B'); 02/05/23
13900  when vbank = 4;                                       02/05/23
14000  json_SetStr      (pJson : 'BusinessUnitDimemsion' : 'PROJ'); 02/05/23
14100  when vbank = 5;                                       02/05/23
14200  json_SetStr      (pJson : 'BusinessUnitDimemsion' : 'CCH'); 02/05/23
14300  when vbank = 6;                                       02/05/23
14400  json_SetStr      (pJson : 'BusinessUnitDimemsion' : 'Export'); 02/05/23
14500  when vbank = 97;                                       02/05/23
14600  json_SetStr      (pJson : 'BusinessUnitDimemsion' : 'Franse VT'); 02/05/23
14700  other;                                              02/05/23
14800  json_SetStr      (pJson : 'BusinessUnitDimemsion' : ''); 02/05/23
14900  ends1;                                              02/05/23
15000  else;                                                 28/04/23
15100  json_SetStr      (pJson : 'BusinessUnitDimemsion' : ''); 28/04/23
15200  endif;                                                28/04/23
15300  else;                                                 28/04/23
15400  json_SetStr      (pJson : 'BusinessUnitDimemsion' : ''); 28/04/23
15500  endif;                                                28/04/23
15600
15700  // pfstafa velden
15800  setll (kfakmr) lfstafaf;                             07/06/23
15900  reade (kfakmr) lfstafaf;                             13/06/23
```

SOURCE FILE CDJSON/QRPGLESRC

```
SOURCE FILE : . . . . . SOURCEID: QR  
MEMBER      : . . . . . JSONINV01
```

SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8 ...+... 9 ...+... 0	
16000 dow not %eof(lfstafaf);	13/06/23
16100 ptemp = json_NewObject();	13/06/23
16200 json_SetInt (ptemp : 'SSTUKS' : SSTUKS);	13/06/23
16300 json_SetNum (ptemp : 'SHOEV' : SHOEV);	13/06/23
16400 json_SetNum (ptemp : 'SVERKW' : SVERKW);	13/06/23
16500 json_SetNum (ptemp : 'SKOMMD' : SKOMMD);	13/06/23
16600 json_SetNum (ptemp : 'SPLATS' : SPLATS);	13/06/23
16700 json_SetNum (ptemp : 'SKOMM' : SKOMM);	13/06/23
16800 json_SetNum (ptemp : 'SPRIV' : SPRIV);	13/06/23
16900 json_SetInt (ptemp : 'SARTGR' : SARTGR);	13/06/23
17000 json_SetInt (ptemp : 'SORDER' : SORDER);	13/06/23
17100 json_SetValue (pJson : 'pfstafaLines[]' : ptemp);	13/06/23
17200 json_delete(pTemp);	13/06/23
17300 reade (kfaknr) lfstafaf;	13/06/23
17400 enddo;	13/06/23
17500 // pfstafa velden	07/06/23
17600	07/06/23
17700 json_SetValue (pNewObj : 'Invoices[]' : pjson);	16/04/23
17800 json_delete(pjson);	21/03/23
17900 reade (kimpda) lffakinv;	16/04/23
18000 enddo;	21/03/23
18100 if teller > 0;	16/04/23
18200 json_WriteJsonStmf(pNewObj: '/export/inv_' + pfirma + '_' + pdatum + '.json': 1208: *OFF);	16/04/23
18300 endif;	16/04/23
18400 json_delete(pNewObj);	21/03/23
18500 *inlr = *on;	06/03/23

* * * * E N D O F S O U R C E * * * *