# SPExtract.ibeblock

```
execute ibeblock ExtractProcedures (
  CodeDir varchar(1000) = 'E:\IBEBlocks\' comment 'Path to necessary
IBEBlocks',
  CreateAlter varchar(6) = 'CREATE',
  Dialect smallint = 3,
  EmptyBody Boolean = FALSE,
  FileStrm variant)
as
begin
  CRLF = ibec_CRLF;
  WriteDDLBlock =
     'execute ibeblock (sName variant, sDDL variant, sInParams variant,
sOutParams variant, sSrc variant, FS variant)
  as
     CRLF = ibec_CRLF();
     if (sInParams <> '''') then
        sDDL = sDDL || '' ('' || CRLF || '' '' || ibec_Trim(sInParams) ||
'')'';
     if (sOutParams <> '''') then
        sDDL = sDDL || CRLF || ''RETURNS ('' || CRLF || '' '' ||
ibec_Trim(sOutParams) || '')'';
     sDDL = sDDL || CRLF || ''AS'' || CRLF;
     sDDL = sDDL || sSrc || ''^'';
     ibec_progress(''Writing procedure '' || sName);
     ibec_fs_Writeln(FS, sDDL); ibec_fs_Writeln(FS, '''');
ibec_fs_Writeln(FS, '''');
  end';

  RdbPrecisionExists = TRUE;
  FldTypeFunc = ibec_LoadFromFile(CodeDir || 'FldType.ibeblock');

  sName = ''; sDDL = ''; sInParams = ''; sOutParams = ''; sParam = ''; iPrec
= 0;
  if (FileStrm is not null) then
     FS = FileStrm;
  else
     FS = ibec_fs_OpenFile('E:\BlockScript.sql', __fmCreate);

  Stmt = ibec_Concat(
  'select pr.rdb$procedure_name,    ', CRLF,  -- 0
  '       pp.rdb$parameter_name,    ', CRLF,  -- 1
  '       pp.rdb$parameter_type,    ', CRLF,  -- 2
  '       fs.rdb$field_name,        ', CRLF,  -- 3
  '       fs.rdb$field_type,        ', CRLF,  -- 4
  '       fs.rdb$field_length,      ', CRLF,  -- 5
  '       fs.rdb$field_scale,       ', CRLF,  -- 6
  '       fs.rdb$field_sub_type,    ', CRLF,  -- 7
```

```
  '       fs.rdb$segment_length,    ', CRLF,  -- 8
  '       fs.rdb$dimensions,        ', CRLF,  -- 9
  '       cr.rdb$character_set_name,', CRLF,  -- 10
  '       co.rdb$collation_name,    ', CRLF,  -- 11
  '       pp.rdb$parameter_number,  ', CRLF,  -- 12
  '       fs.rdb$character_length,  ', CRLF,  -- 13
  '       fs.rdb$default_source     ', CRLF); -- 14

  if (not EmptyBody) then
     Stmt = ibec_Trim(Stmt) || ',' || CRLF || ' pr.rdb$procedure_source' ||
CRLF;
  else
     sSrc = 'BEGIN' || CRLF || ' EXIT;' || CRLF || 'END';
  if (RdbPrecisionExists) then
     Stmt = ibec_Trim(Stmt) || ',' || CRLF ||
            '        fs.rdb$field_precision' || CRLF;
  Stmt = Stmt ||
  'from rdb$procedures pr'              || CRLF ||
  'left join rdb$procedure_parameters pp on pp.rdb$procedure_name =
pr.rdb$procedure_name' || CRLF ||
  'left join rdb$fields fs on fs.rdb$field_name = pp.rdb$field_source'
|| CRLF ||
  'left join rdb$character_sets cr on fs.rdb$character_set_id =
cr.rdb$character_set_id'   || CRLF ||
  'left join rdb$collations co on ((fs.rdb$collation_id =
co.rdb$collation_id) and'        || CRLF ||
  '                                (fs.rdb$character_set_id =
co.rdb$character_set_id))'   || CRLF ||
  'order by pr.rdb$procedure_name, pp.rdb$parameter_type,
pp.rdb$parameter_number';

  SetTermWritten = FALSE;

  for execute statement :Stmt into :SPProps
  do
  begin
     if (SetTermWritten = FALSE) then
     begin
        ibec_fs_Writeln(FS, 'SET TERM ^ ;' || CRLF);
        SetTermWritten = TRUE;
     end;
     if (RdbPrecisionExists = TRUE) then
        iPrec = ibec_IIF(EmptyBody = 1, SPProps[15], SPProps[16]);

     SPName = ibec_Trim(SPProps[0]);
     if (sName <> SPName) then
     begin
        if (sDDL <> '') then
           execute ibeblock WriteDDLBlock(sName, sDDL, sInParams,
sOutParams, sSrc, FS);
```

```
        sName = SPName;
        if (not EmptyBody) then
           sSrc = ibec_Trim(SPProps[15]);
        sDDL = CreateAlter || ' PROCEDURE ' || SPName;
        sInParams = ''; sOutParams = ''; sParam = '';
     end
     if (SPProps[1] is not null) then
     begin
        execute ibeblock FldTypeFunc(SPProps[4], SPProps[7], SPProps[5],
SPProps[6], SPProps[8],
                                     SPProps[13], SPProps[16], Dialect)
                returning_values :sParam;
        sParam = ibec_Trim(SPProps[1]) || ' ' || sParam;
        -- Character Set
        if ((SPProps[4] in (14, 37, 261)) and (SPProps[10] is not null))
then
           sParam = sParam || ' CHARACTER SET ' || ibec_trim(SPProps[10]);
        -- Default Value
        if ((SPProps[14] is not null) and (SPProps[14] <> '')) then
           sParam = sParam || ' DEFAULT ' || ibec_trim(SPProps[14]);
        if (SPProps[2] = 0) then
        begin
           if (sInParams <> '') then
              sInParams = sInParams || ',' || CRLF || ' ';
           sInParams = sInParams || sParam;
        end
        else if (SPProps[2] = 1) then
        begin
           if (sOutParams <> '') then
              sOutParams = sOutParams || ',' || CRLF || ' ';
           sOutParams = sOutParams || sParam;
        end
     end
  end
  if (sDDL <> '') then
     execute ibeblock WriteDDLBlock(sName, sDDL, sInParams, sOutParams,
sSrc, FS);

  if (SetTermWritten) then
     ibec_fs_Writeln(FS, 'SET TERM ; ^' || CRLF);

  if (FileStrm is null) then
     ibec_fs_CloseFile(FS);
end
```

From:
https://www.ibexpert.com/docu/ - **IBExpert**

Permanent link:
**https://www.ibexpert.com/docu/doku.php?id=06-ibexpert-ibeblock-examples:extract-metadata-using-ibeblock:spextract**

Last update: **2023/04/28 02:27**