Tartalom:

1. Általános működés
2. Részletes működés
3. Debuggolás

# Általános működés

A program célja, hogy a Kulcs-Ügyvitel szoftverben levő Bizonylat export napi rendszerességgel fusson, majd az eredmény export fájl a SharePointban a megfelelő mappába kerüljön.

A program elkészítéséhez használt technológiák:

* node.js
* firebird sql
* powershell

A program fájljai a dataplexben lévő Ipsolos szervergépén található a C:\KS\_fajlok\node-test helyen.

# Részletes működés

A program rendszeres futása úgy valósul meg, hogy a windows feladatütemezőben van egy KS\_Projekt\_status\_export nevű automatizmus, amely napi rendszerességgel elindítja a node scriptet.

A node script felépítése a következő:

1. adatbázis autentikációs fájl másolása
2. adatbázis lekérések
3. lekért adatok excelbe írása
4. kigenerált excel feltöltése a SharePoint rendszerébe

A fentebbi lépések részletesen.

1. Adatbázis autentikációs fájl másolása

Amikor a KulcsSoft oktató embere beüzemelte a firebird elérést, akkor azt mondta, hogy minden alkalommal amikor a Kulcs-Ügyviteli rendszer frissül a gépen, akkor újra felül kell írni a C:/ProgramData/KS/FbDatabaseServer/security2.fdb fájlt azzal a security2.fdb fájllal, ami tartalmazza ami lokális elérési jogosultságainkat.

Ezt funkciót a copyDBFile függvény hajtja végre.

1. Adatbázis lekérések

Az adatbázis queryk teszteléséhez és az adatbázis szerkezetének megértéséhez telepítették a gépre a DBeaver programot.

Az adatbázisba bejelentkezéshez szükséges információk:

host: localhost  
port: 3056  
acc: bubment  
pw: Ips0l2020  
db:C:/ProgramData/KS/FbDatabaseServer/Databases/KSCOMPANY\_IPSOLRENDSZERHZZRT\_191114155447.KSFDB

Az oktatás során elmondták, hogy a számlákkal kapcsolatos információk a CustomerStockOut és a CustomerStockOutDetail táblában, míg a vevői rendelésekkel kapcsolatos adatok a CustumerOrder és a CustomerOrderDetail táblában találhatóak

(Kieg: A Kulcs-Soft felületéről exportálható „Bizonylatok tételes lekérdezése.xlsx” fájlban található celláknál ahol a „Bizonylat fajta” oszlopban a Vevői rendelés látható az az adatbázisban a Teljesítési igazolásnak fajtájú bizonylatnak (VoucherSequence) felel meg annak ellenére, hogy az adatbázisban létezik Vevői rendelés fajtájú bizonylat.)

Az SQL queryk intuitív módon készültek az adatbázis és a „Bizonylatok tételes lekérdezése.xlsx” összehasonlításával.

A node-firebird npm package-val volt egy olyan probléma, hogy a package új verziójának futtatása során timeout nélkül végtelen ideig kérdezett le a query. Ha ez a probléma újra fellép, akkor a node-firebird github oldalán az issueknál érdemes először keresni a megoldást.

A query funkciókat a firebirdQueries függvény hajtja végre.

A query funkciók futtatása és a queryk eredményének mentése úgy lett megoldva, hogy esetlegesen szükséges új queryk esetén könnyen bővíthető legyen a program.

Ha esetlegesen az adatbázis tartalmával bővíteni szeretnénk az excel exportot annak a menete a következő:

* 1. Bővítjük az szükséges query-t a queryInfo tömbben, úgy hogy lekérdezzen még egy típusú oszlopot.
  2. Kibővítjük egy új elemmel a columnInfo tömböt az alábbi módon:
     1. az elem ColumnHU értéke a kigenerált „Bizonylatok tételes lekérdezése.xlsx” fájlban levő oszlop fejléce legyen
     2. az elem ColumnEN értéke az adatbázis lekérés oszlopfejléce legyen
     3. az elem ColumnNumer értéke pedig az legyen ahányadik oszlopban az adott értékben szerepelnie kell az excel exportban.

1. Lekért adatokból excel kigenerálása

Az export funkciót a writeExcel függvény hajtja végre.

(Kieg: Ez az egyetlen függvény, ami nincs felkészítve, arra, hogy bővülhet a program)

1. Kigenerált excel feltöltése a SharePointba

A program ezen része PowerShell technológiákat használ. A program esetleges költöztetése esetén. Olvasd el a script\shell\init mappában található README fájlt.

A PowerShell scripteket a runShellScripts függvény futtatja.

A Shell scriptek futtatásához szükséges információk a shellParams tömbben találhatóak.

A scriptek futásának menete a következő:

* A shellParams aktuális elemének psFile értéke a futtatandó powershell script fájl neve kiterjesztés nélkül
* a params altömb tartalmazza az aktuálisan futtatandó scriptek bemeneti paramétereit

A powershell script futtatásához le van tárolva egy titkosított jelszó a script\shell\creds mappában. Ha megváltozott a cégnél a [tavmeres@ipsol.hu](mailto:tavmeres@ipsol.hu) fiókhoz tartozó jelszó, akkor futtasd a pass\_change.ps1 programot és kövesd az utasításait.

# Debuggolás

Ha a program esetleg hibára futna akkor a futtatási környezetében a „node index” parancs futtatása után a consol logokból meg lehet állapítani, hogy hol akad el a program.