

TECHNICKÁ UNIVERZITA V KOŠICIACH
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Import dát z produkčných systémov do RDBS Oracle
Prípadová štúdia

Školiteľ: doc. Ing. Ján Jadlovský, CSc.
Konzultant: Ing. Tomáš Tkáčik

2023 Košice

Bc. Samuel Oklamčák

Obsah

1. Štruktúra štúdie.....	3
1.1. Cieľ.....	3
1.2. Otázky.....	3
1.3. Predpoklady	4
1.4. Prípravná etapa	4
1.5. Etapa importu a exportu	5
1.6. Príklady.....	8
1.6.1. Príklad 1 – import dát pomocou tabuľkových súborov.....	8
1.6.2. Príklad 2 – MySQL – použitie SQL na import dát.....	9
1.6.3. Príklad 3 – MySQL – import pomocou migračného repozitára.....	11
1.6.4. Príklad 4 – MSSQL – export dát do tabuľkových súborov	15
1.6.5. Príklad 5 – MSSQL – vlastný skript	17
1.6.6. Príklad 6 – MariaDB – export cez phpMyAdmin	19
2. Dodatočné informácie.....	22
2.1. Nastavenie prostredia MySQL Workbench	22
2.2. Konektory tretích strán v programe Oracle SQL Developer.....	23
2.2.1. Získanie knižnice.....	24
2.2.2. Overenie funkčnosti	26
2.3. Nastavenie prostredia Microsoft SQL	27
2.4. Prevod JSON dát do SQL.....	28
Záver.....	30

1. Štruktúra štúdie

V tejto štúdii rôzne spôsoby získania dát z iných databázových systémov, ktoré môžu pomôcť pri tvorbe MIS. Neopisujeme priamo produkčné informačné systémy, ako napr. CRM (SuiteCRM, Odo), ERP (NetSuite, SAP S/4HANA) alebo fakturačné systémy (InvoiceNinja, QuickBooks), pretože týchto systémov je veľmi veľké množstvo. Zamerali sme sa na omnoho dôležitejší fakt, a to ten, že každý z týchto systémov ukladá dáta do databázy. Popisujeme hlavne MySQL, MariaDB a MSSQL. Databázových systémov je mnoho, dôležité je ukázať **postup pri práci**. Študenti si musia naštudovať používanú databázu, z ktorej budú získavať dáta.

Názov	Import dát z produkčných systémov do RDBS Oracle
Aktuálny stav	Pri výučbe predmetu je používaný ako ukážkový import dát zo súborov typu Comma Separated Values
Cieľ	Vytvoriť študijné materiály zaoberajúce sa importom dát z rôznych databázových systémov, bez ohľadu na produkčný informačný systém
Riešenie	Presne definovať potrebné predpoklady a znalosti na vykonanie tohto kroku počas tvorby MIS. Vytvoriť flow diagram, ktorý všeobecne opisuje proces importu dát, a vytvoriť sériu príkladov ako ukážku tohto procesu
Body realizácie	<ol style="list-style-type: none"> 1. návrh a definícia potrebných predpokladov pre vykonanie procesov importu dát 2. návrh a tvorba flow diagramu postupu 3. výber databázových systémov 5. návrh a tvorba príkladov

Tab. 1 Štruktúra PŠ 1

1.1. Cieľ

Na predmete Manažérske Informačné systémy, používame databázu od spoločnosti Oracle. K správe tejto databázy budeme používať Oracle SQL Developer. V tomto nástroji budeme vytvárať tabuľky, vkladať záznamy, aktualizovať záznamy alebo upravovať ich. Vkladanie dát do databázy sa skladá z viacerých krokov a existuje viacero spôsobov. Dáta môžu pochádzať z csv, sql, xml, json alebo iných súborov. Cieľom tejto štúdie je rozšíriť znalosti v oblasti vkladania dát do databázového systému Oracle.

1.2. Otázky

- Čo potrebujeme od zákazníka, aby sme doručili požadované výstupy?
- Aké informačné systémy používa zákazník, pre ktorého ideme vypracovať MIS?
- Aké je zloženie dát, preferované merné jednotky, názvy lokalít?

1.3. Predpoklady

Riešiteľ, resp. študent potrebuje od zákazníka prístup k databázovým systémom, potrebuje sa oboznámiť s formou uloženia dát, prevádzkovými časmi a potrebuje zistiť štruktúru dát (dopytovanie zákazníka).

Pred importom je potrebné vybrať spôsob importu a tomu prispôbiť proces prípravy. Pri tvorbe vlastného skriptu je potrebné nájsť vhodný programovací jazyk, nájsť aplikačné knižnice pre zdrojovú a cieľovú databázu, overiť ich funkčnosť, vybrať spôsob dočasného uloženia dát, definovať potrebné transformácie alebo úpravy dát.

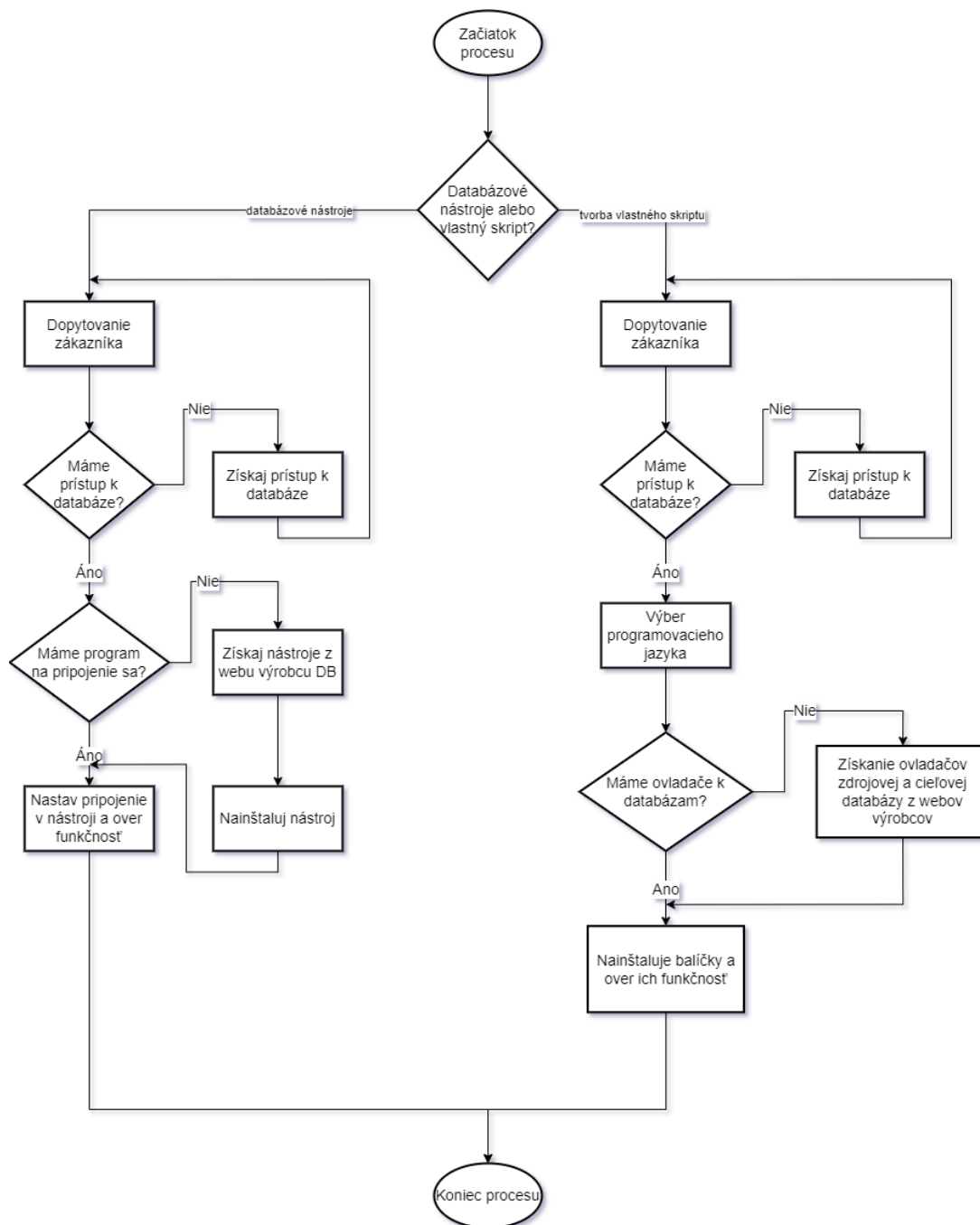
Ak budeme pracovať s nástrojmi priamo od výrobcu databáz, máme možnosť vybrať nástroje s grafickým alebo bez grafického prostredia. Celý proces importu dát sme rozdelili na tieto etapy:

1. prípravná etapa, v ktorej zbierame informácie od zákazníka a vyberáme spôsob importu dát,
2. etapa importu a exportu, ktorá popisuje kroky potrebné na exportovanie dát, dočasné uloženie, resp. upravenie dát a vloženie dát do cieľovej databázy.

V druhej etape je možné kombinovať tieto procesy medzi sebou. Druhá etapa má štyri rôzne cesty. Dáta vieme importovať pomocou CSV súborov, SQL skriptu, migračného repozitára alebo vlastného programu v ľubovoľnom programovacom jazyku.

1.4. Prípravná etapa

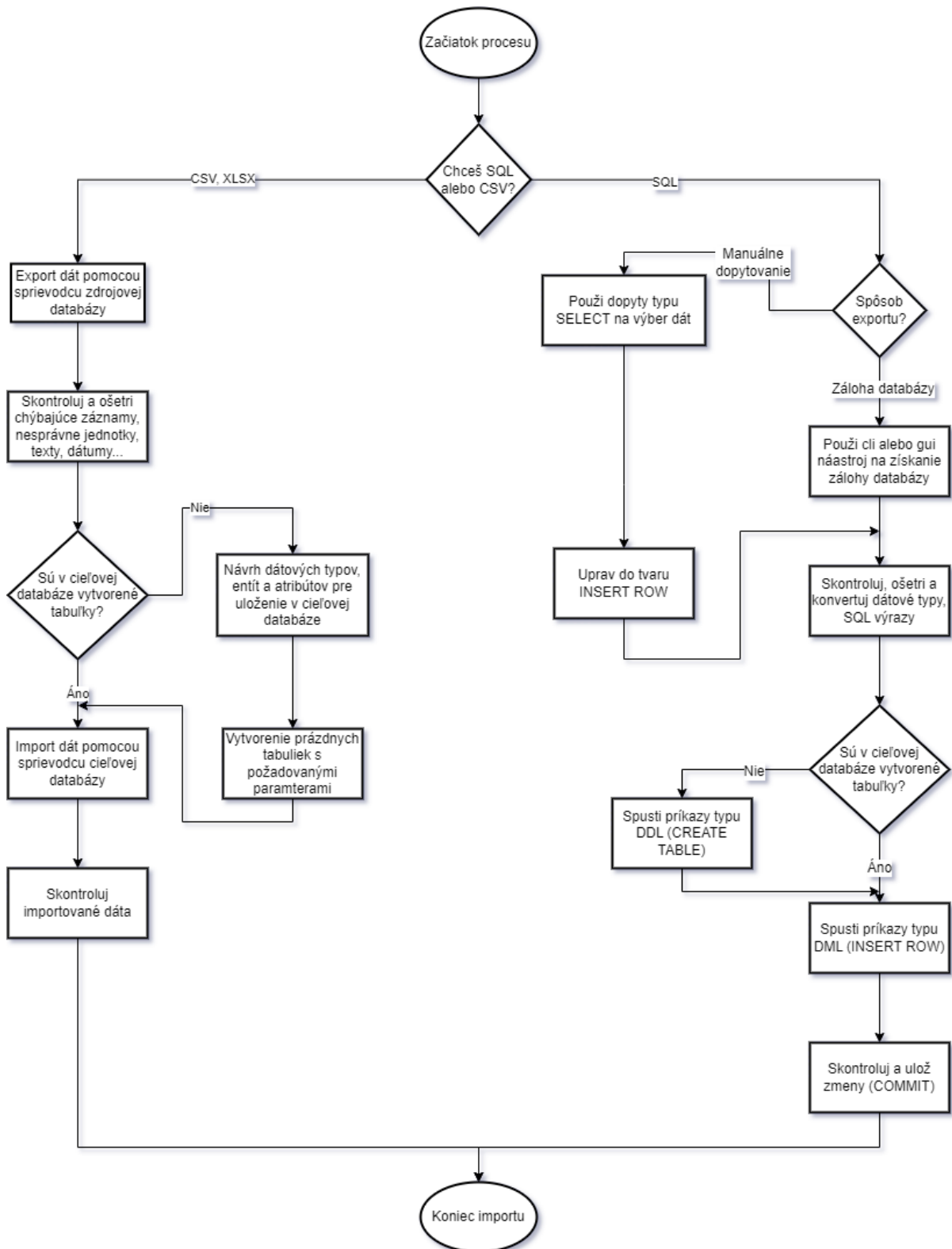
Túto etapu sme popísali pomocou vývojového diagramu, ktorý obsahuje všeobecný postup pri tomto procese. Je možné ho rozdeliť na import dát pomocou databázových nástrojov a na import dát pomocou externého programu, resp. vlastného skriptu. Pozri Obr. 1, rozhodni sa akou cestou sa vyberieš a nájsi správne informácie a nástroje.



Obr. 1 Prípravná etapa

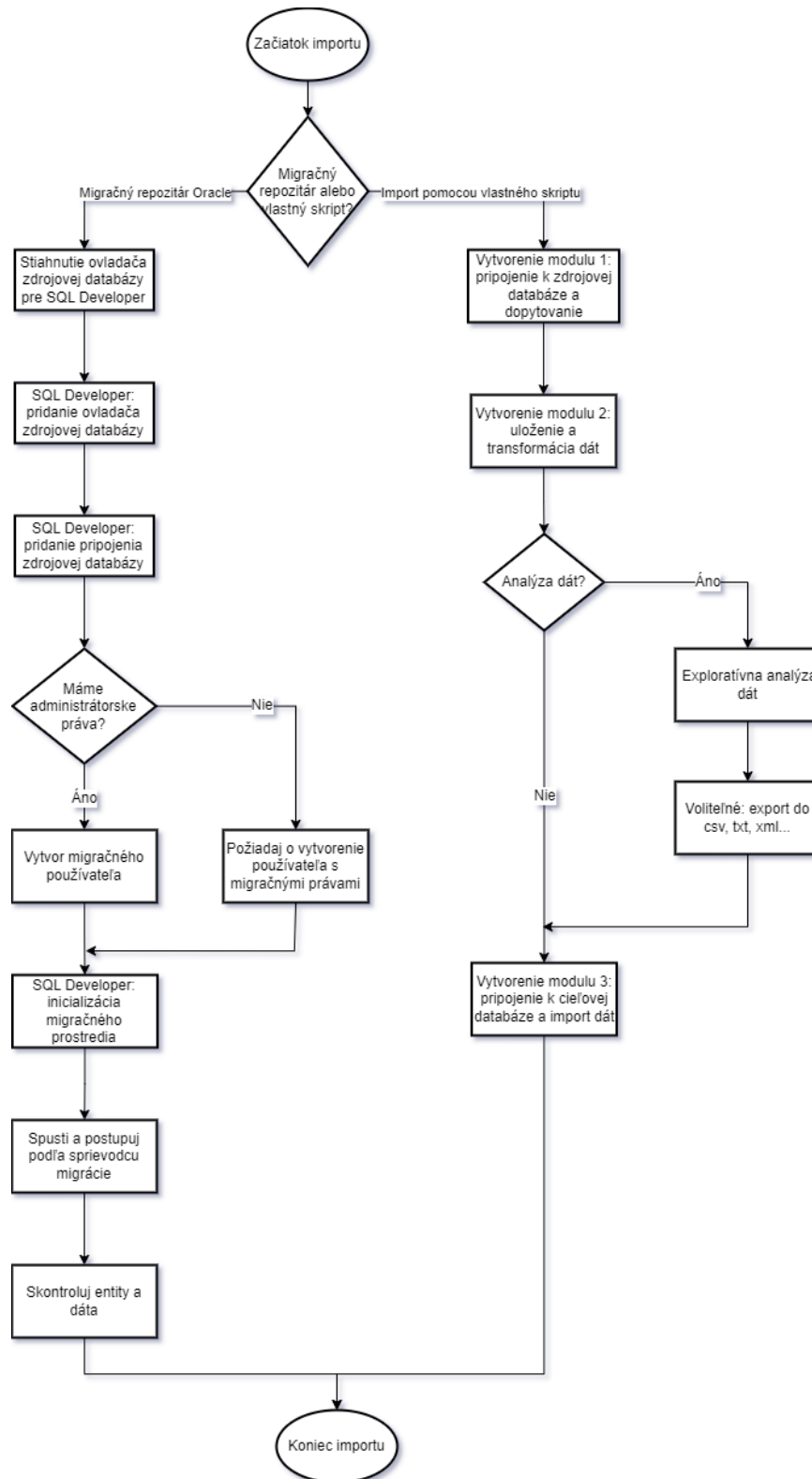
1.5. Etapa importu a exportu

Po výbere správneho smeru, je na výber viacero možností. Prvá možnosť je import dát cez tabuľkové súbory. Druhá možnosť je získať a upraviť dáta cez SQL. Táto možnosť je zložitejšia. Tretia možnosť použije tzv. migračný proces databázy. Je vhodná ak je zložitosť údajov komplikovanejšia a ručné pretypovanie napr. SQL by bolo zložité. Štvrtá možnosť používa ľubovoľný programovací jazyk na získanie, spracovanie (uloženie) a načítanie dát do databázy Oracle.



Obr. 2 Etapa exportu a importu – tabuľkové súbory alebo SQL

Na Obr. 3 sa nachádza proces pre prvú a druhú možnosť získania, úpravy a načítania dát.



Obr. 3 Etapa exportu a importu – migračný repozitár alebo skript

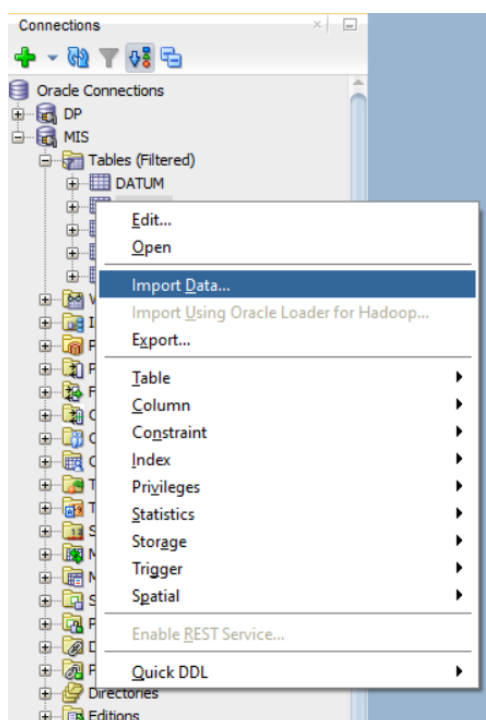
Na Obr. 3 je na výber migračný repozitár alebo vlastný skript. Migračný repozitár vyžaduje administrátorské oprávnenia, resp. vytvorenie migračného používateľa a migračného repozitára. Skript ale umožňuje mať úplnú kontrolu nad dátami a spraviť exploratívne analýzy.

1.6. Príklady

Táto časť obsahuje vytvorenú sadu príkladov, ktorá aplikuje vývojové diagramy a ukazuje procesy importu dát z viacerých databáz.

1.6.1. Príklad 1 – import dát pomocou tabuľkových súborov

Tento príklad je import dát priamo z csv alebo xlsx. Niektoré databázové systémy umožňujú exportovať záznamy vo forme vyššie spomenutých typov súborov. Najprv sa potrebujeme pripojiť k svojej databáze pomocou programu Oracle SQL Developer. Nájdeť modul Connections, resp. ho vyvoláme cez horné menu **View -> Connections**. Následne si **nájdeť tabuľku** do ktorej chceme importovať dáta. **Pravým tlačidlom vyvoláme menu** a klikneme na **Import Data**.



Obr. 4 Príklad 1 – spustenie sprievodcu importu dát

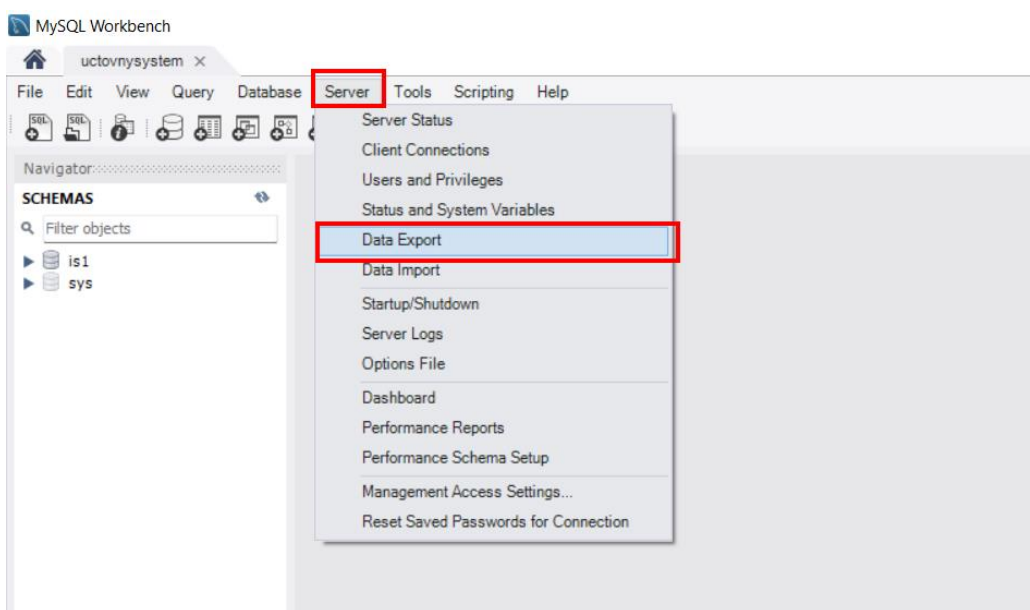
Otvorí sa sprievodca. V prvom kroku si cez tlačidlo **Browse** nájdeme a vyberieme súbor. Uvidíme ukážku dát, ak je potrebné upravíme formát, hlavičku, oddeľovač a ďalšie. Po kliknutí na tlačidlo **Ďalej**, v Import Method necháme Insert, poprípade nastavíme koľko riadkov chceme vložiť. Pokračujeme cez tlačidlo **Ďalej**, následne si vyberieme, ktoré atribúty chceme vybrať. Ak nič nemeňme pokračujeme tlačidlom **Ďalej**. V sprievodcovi (krok č. 4) priradzujeme atribúty zo súboru atribútom v databázovej tabuľke. Ak má súbor hlavičku, sprievodca dokáže nájsť zhodu v názve atribútov a priradiť ich. Po prechode na ďalší krok sprievodcu (krok č. 5) uvidíme výsledný sumár operácie a informácie o procese. Samotný proces importu dát spustíme cez tlačidlo **Dokončiť**.

1.6.2. Príklad 2 – MySQL – použitie SQL na import dát

V tomto príklade importujeme dáta pomocou SQL skriptu. Postup ilustrujeme na databáze MySQL. Vyžaduje používateľský účet v zdrojovej databáze s oprávneniami na čítanie zdrojových údajov. K databáze môžeme pristupovať pomocou CLI a spustiť program **MySQLdump**. Tento nástroj dokáže exportovať celú databázu ako SQL skript, ktorý je možné spustiť a nahráť celú databázu. Umožňuje aj export dát do samotných csv alebo xml súborov. Vieme nastaviť zálohovanie dát pre celý systém (viacero databáz), jednu databázu alebo množinu tabuliek z databázy. Máme tak možnosť prebrať len tabuľku objednávok z produkčného systému, ktorý obsahuje aj tabuľky zákazníkov, faktúr, adries a pod. Náročnosť môže byť pre niekoho vyššia. Omnoho jednoduchšia je práca v nástrojoch, kde môžeme pomocou GUI exportovať dáta.

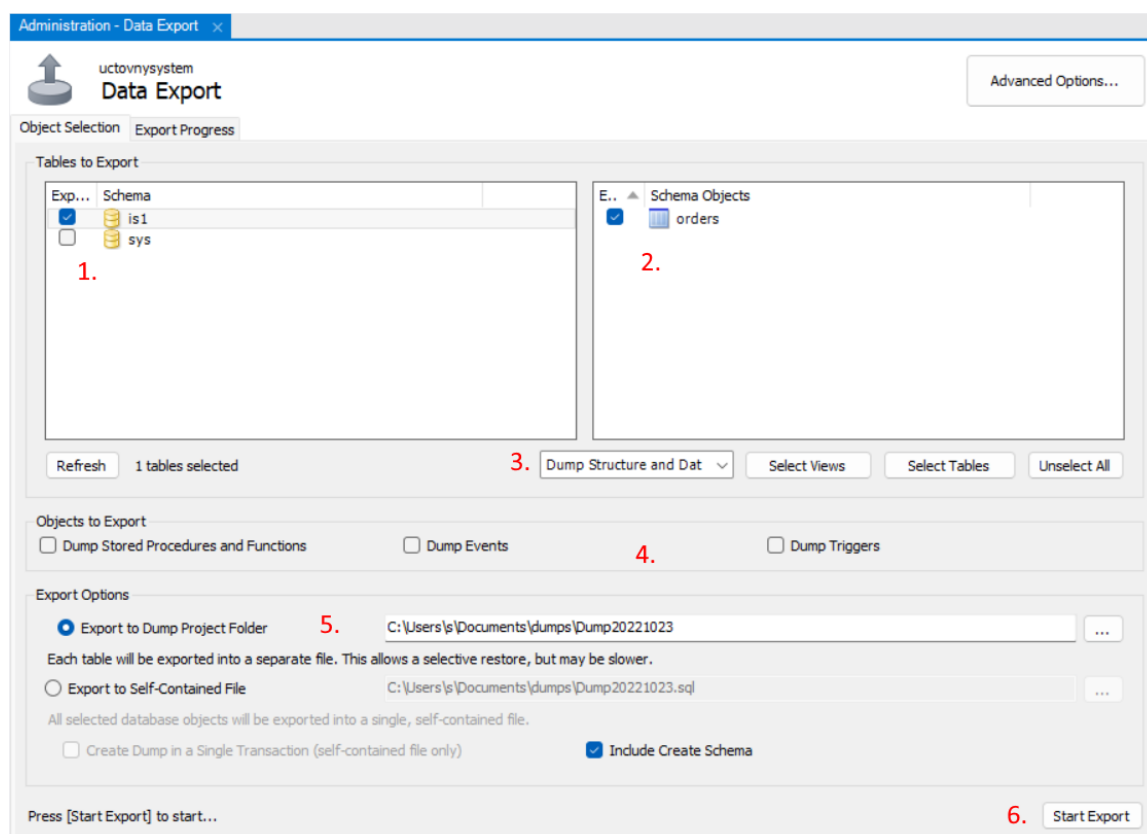
Medzi GUI nástroje podobné Oracle SQL Developer patria MySQL Workbench, Microsoft SQL Server Management Studio, DBeaver alebo dbForge Studio. V závislosti od produkčného systému môžeme mať k dispozícii aj webové nástroje ako phpmyadmin alebo Adminer. Všeobecne by sme mohli povedať, že v nástroji potrebujeme nájsť funkčnosť **Export Data** alebo **Generate SQL**. Zobrazí sa sprievodca, kde by sme mali mať možnosť vybrať množinu tabuliek, celú databázu, pohľady, spúšťače a funkcie.

Použijeme MySQL Workbench ako ukážku sprievodcu na export dát. Najprv je potrebné pridať pripojenie k zdrojovej databáze. Potom sa pripojíme k databáze. V hornom menu klikneme na záložku **Server** a v zozname nájdeme **Data Export**.



Obr. 5 Príklad 2 – spustenie sprievodcu exportu v programe MySQL Workbench

Otvorí sa sprievodca na export dát. Vyberieme databázu **(1)**, tabuľky **(2)**, čo všetko chceme exportovať napr. dáta a štruktúru **(3)**, spúšťače a funkcie **(4)**. Následne vyberieme spôsob ako chceme uložiť export. Máme možnosť vytvoriť priečinok s viacerými skriptmi alebo vložiť všetko do jedného **(5)**. Možnosť Include Create Schema použijeme ak ešte nemáme vytvorenú samotnú databázu. Posledný krok je spustenie exportu **(6)** kliknutím na **Start Export**. V lokalite definovanej v Export Options **(5)** nájdeme SQL skript.



Obr. 6 Sprievodca pre nastavenie parametrov exportu

Možnosť **Include Create Schema**, pridá do SQL skriptu nižšie zvýraznený výraz, ktorý môžeme odstrániť, resp. zakomentovať aby sa nespustil.

```
CREATE DATABASE IF NOT EXISTS `is1`;  
USE `is1`;
```

Skript 1 – Výraz pre vytvorenie databázy

Vygenerovaný SQL skript, bude odlišný od používaného PL/SQL v Oracle databáze. Zmeny dátových typov alebo iných výrazov je možné upraviť ručne alebo pomocou online konvertora (sqlines.com/online). Konvertor potrebuje definovať zdrojový a cieľový DBMS a potom vložiť SQL skript. Aj po konverzií je niekedy potrebné ručne opravovať výrazy, ktoré sú špecifické pre staršie verzie a nie sú podporované v novej verzii alebo majú zmenenú syntax. Tento typ importu vyžaduje viac znalostí jazyka SQL. Veľmi častým problémom pri migrácii dát cez SQL medzi databázami sú

atribúty obsahujúce dátum a čas. Je potrebné dočasne upraviť napr. dátumové formáty. Počas importu dát sa následne konvertujú na TIMESTAMP a formát používaný cieľovou databázou. Príklad zmeny formátu TIMESTAMP pre kompatibilitu s MySQL formátom (yyyy-mm-dd) vidíme nižšie.

```
alter session set nls_timestamp_format='YYYY-MM-DD HH24:MI:SS' ;
```

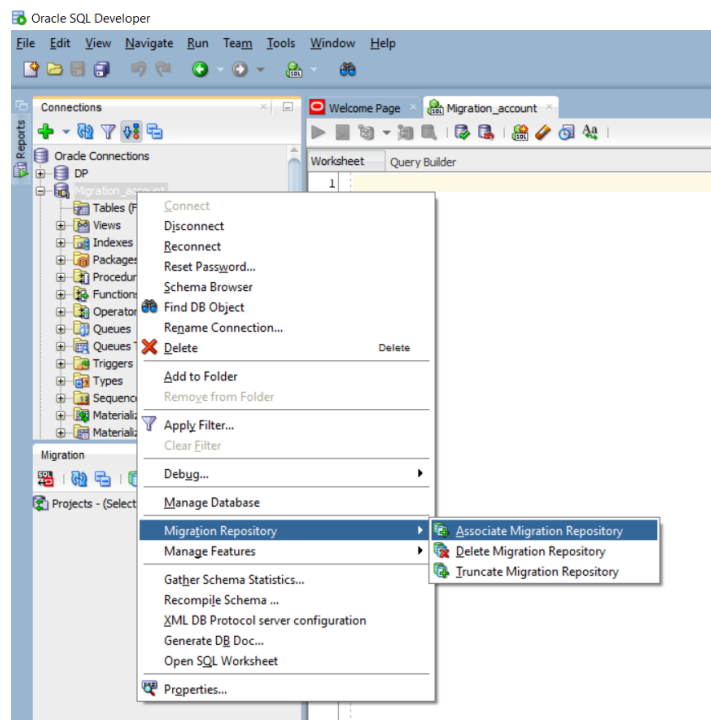
Skript 2 – Výraz pre zmenu formátu dátumu a času v Oracle

Následne sa presunieme do nástroja Oracle SQL Developer, pripojíme sa k cieľovej databáze. Cez horné menu **File -> Open** otvoríme SQL skript. Skript sa otvorí v tzv. SQL Worksheet, kde vieme upravovať a spustiť skript. Vkladanie záznamov cez INSERT sa ukončuje príkazom COMMIT, ktorý uloží zmeny. Kontrolu importu vieme vykonať v bočnom menu **Connections**, kde si rozbalíme zoznam **Tables**, nájdeme tabuľku a otvorím jej náhľad. Prepne sa do časti **Data**, kde by sme mali vidieť vložené údaje.

1.6.3. Príklad 3 – MySQL – import pomocou migračného repozitára

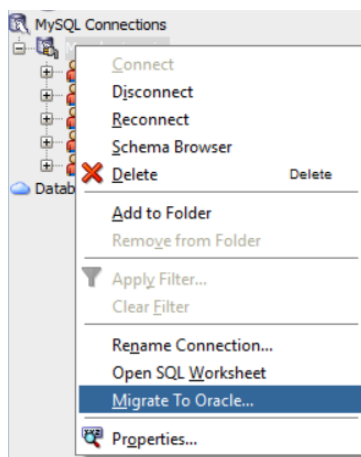
Príklad migrácie dát vyžaduje administrátorské oprávnenia v cieľovej databáze a používateľské oprávnenie v zdrojovej databáze. V tomto postupe sa migruje databáza. Cely proces prebieha na strane cieľovej databázy, resp. v programe Oracle SQL Developer. Do SQL Developera potrebujeme stiahnuť JDBC konektor k MySQL Databáze. Je to knižnica v programovacom jazyku Java vytvorená na pripojenie sa k databáze MySQL, vo vývojovom prostredí. Knižnicu nájdeme na webe výrobcu. V programe SQL Developer prejdeme do **Tools -> Preferences -> Database -> Third Party JDBC Driver**, kde pridáme stiahnutý súbor. V sprievodcovi pridania pripojenia pribudne možnosť MySQL databázy. Pri pridávaní pripojenia vložíme hostname a port, klikneme na Choose Database, obnoví sa zoznam dostupných databáz, kde vyberieme zdrojovú databázu a nakoniec uložíme spojenie.

Pre nasledujúci krok potrebujeme administrátorské oprávnenia. Ak sme administrátor, tak nájdeme na webe postup ako vytvoriť migračného používateľa alebo požiadame administrátora na vytvorenie migračného účtu. Ak dostaneme prihlasovacie údaje k migračnému používateľovi, pridáme ho medzi pripojenia do SQL Developera a prihlásime sa. V paneli pripojenia klikneme na migračného používateľa pravým tlačidlom, nájdeme **Migration Repository** a vyberieme **Associate Migration Repository**.



Obr. 7 Spustenie procesu tvorby migračného repozitára

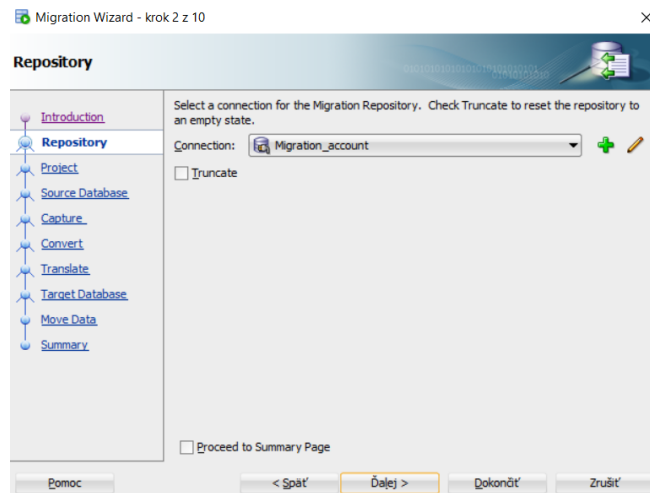
Spustí sa proces tvorby tabuliek, na uloženie dát a metadát o migrácií. Po skončení tohto procesu sa prihlásime do cieľovej MySQL databázy pomocou SQL Developera. V časti **Connection**, klikneme pravým tlačidlom na názov pripojenia a zo zoznamu vyberieme možnosť **Migrate to Oracle**.



Obr. 8 Spustenie migrácie databázy

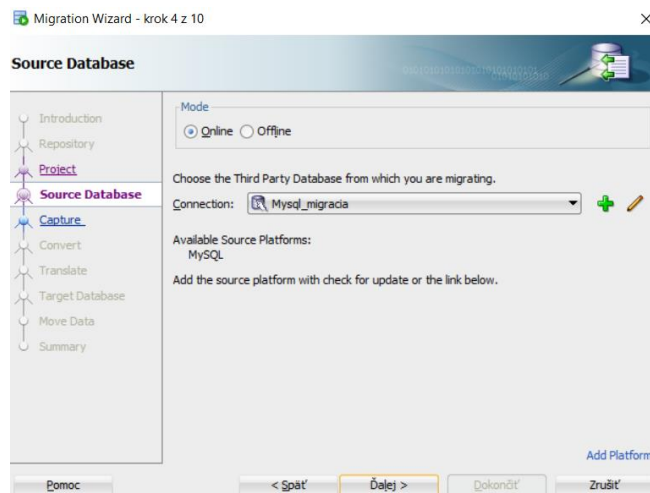
Otvorí sa nové dialógové okno. V sprievodcovi postupujeme nasledovne:

- **Krok 1** - sme informovaní o postupe. Kliknutím na tlačidlo **Ďalej** sa dostaneme na výber migračného používateľa. Políčko **Truncate** nezaškrtneme, keďže je to naša prvá migrácia a nepotrebujeme vymazať obsah tabuliek.



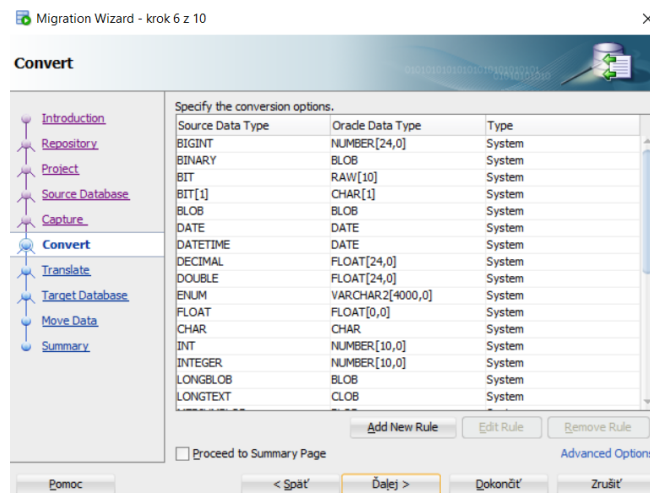
Obr. 9 Sprievodca migráciou – výber migračného repozitára

- **Krok 3** – vkladáme názov a miesto uloženia projektu. Proces je ukladaný ako projekt. V tejto lokalite nájdeme SQL skripty, ktoré vykonali migráciu, a txt log ohľadom prípadných chýb. Nasleduje **krok 4**, kde vyberáme zdrojovú databázu, z ktorej migrujeme dáta. Spôsob migrácie vyberáme **Online**.



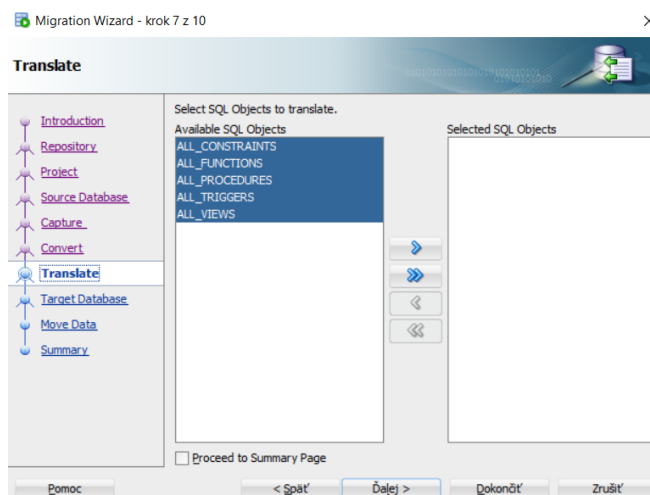
Obr. 10 Sprievodca migráciou – výber zdrojovej databázy

- **Krok 5** - vyberieme, ktoré databázy chceme migrovať. V našom prípade to je databáza is1. **Nasledujúci krok** umožňuje pridávať, upravovať alebo vymazávať konverzie medzi dátovými typmi. Tento krok konvertuje dátové typy, podobne ako sme to robili pri druhom spôsobe pomocou online konvertora.



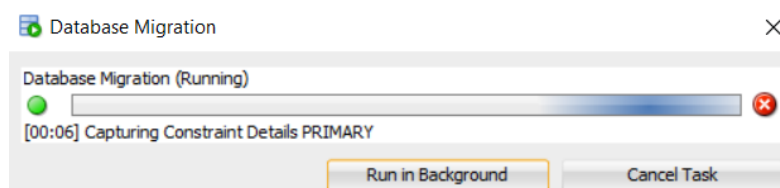
Obr. 11 Sprievodca migráciou – definovanie konverzií dátových typov

- **Krok č. 7** slúži na definovanie objektov databázy, ktoré chceme migrovať. Ak nevyberieme žiadne funkcie, pohľady a iné, tak sa skopírujú len tabuľky s dátami. Takto by sme mohli stratiť funkcionality napr. automatického vkladanie identifikátorov a pod. Vytvorený MIS, bude len čítať dáta (nebude ich upravovať), ale rozhodnutie ponechávame na užívateľovi.



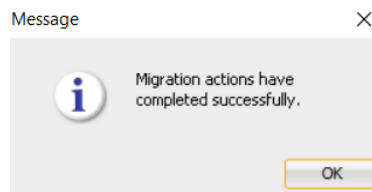
Obr. 12 Sprievodca migráciou – úprava vlastnosti migrácie

- Medzi poslednými krokmi, vyberáme cieľovú databázu, kam sa importujú dáta. Následne skontrolujeme proces presunu dát a stlačíme **Dokončit**. Spustí sa proces migrácie.



Obr. 13 Prebiehajúci proces migrácie

Proces trvá minimálne niekoľko minút. Pri veľkej databáze to môže byť aj niekoľko hodín. Dôležitá je správa o ukončení migrácie. Ak sa zobrazí táto správa, údaje sú uložené v databáze a prebehol aj príkaz COMMIT.



Obr. 14 Správa o úspešnej migrácii

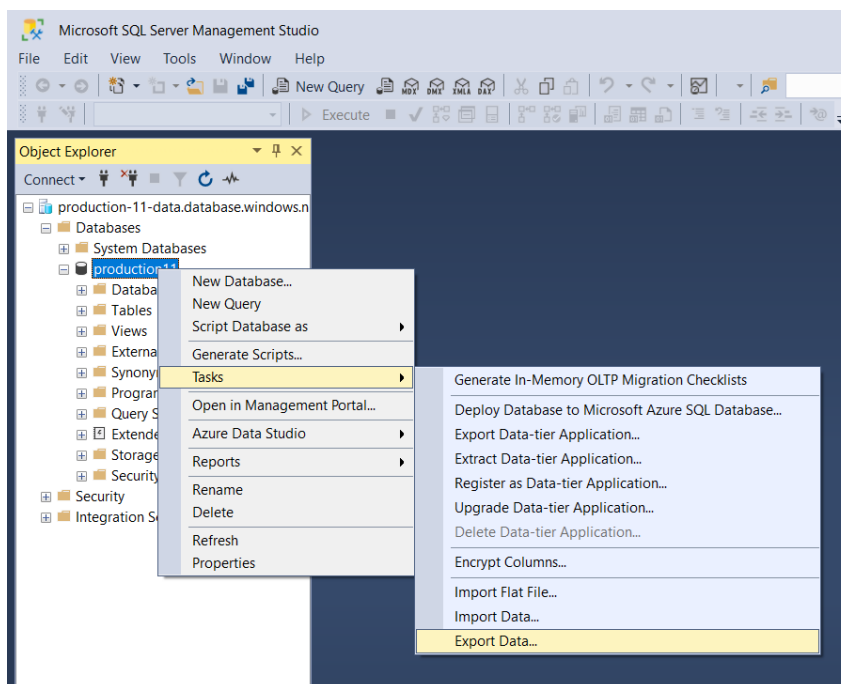
V prípade že sa ukáže chyba ORA-01400 tak máme prázdne hodnoty v dátach, podmienené nenulové atribúty alebo nemáme správne nakonfigurovaného migračného používateľa. Môže sa stať, že migračný používateľ nemá dostatočné oprávnenia.

1.6.4. Príklad 4 – MSSQL – export dát do tabuľkových súborov

Je veľmi podobný prvému, ale s tým rozdielom, že nemáme vstupné csv alebo xlsx súbory. Na ukážkovej databáze Microsoft SQL Server vykonáme sériu krokov aby sme tieto súbory vygenerovali. Pri používaní DBMS od spoločnosti Microsoft máme znovu viacero možností exportu, resp. importu dát. Ako aj iné databázy, aj táto nám umožňuje použiť grafické nástroje na manažment databázy, a ďalšie operácie ako import a export dát. Grafické nástroje určené pre SQL Server od Microsoft sú hlavne SQL Server Management Studio (SSMS) alebo v prípade cloudových služieb napr. Azure Data Studio. V grafickom nástroji máme možnosť exportovať dáta do zaužívaných tabuľkových formátov súborov ako csv, xls alebo iné ako sql skript. K databáze sa vieme pripojiť aj pomocou shellu alebo rôznych ovládačov.

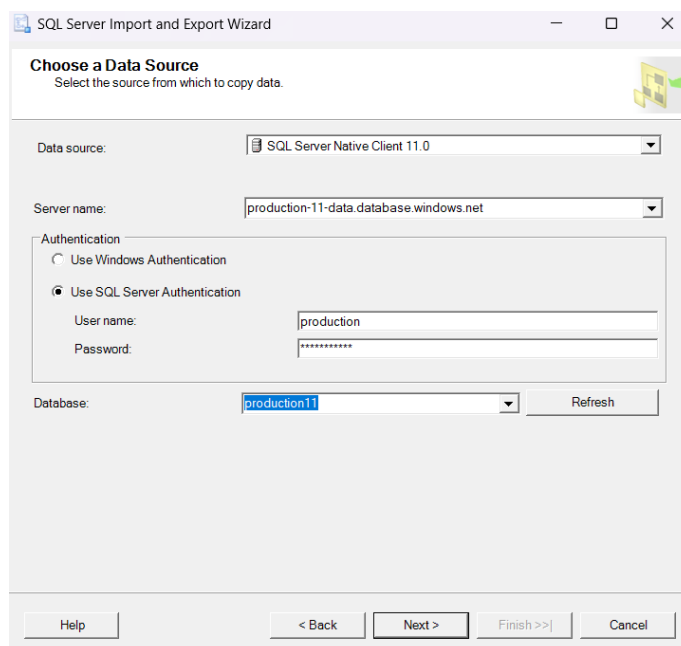
Stiahneme program SQL Server Management Studio zo stránky výrobcu. Nainštalujeme program. Následne si pridáme pripojenie k našej databáze. Po spustení programu sa v strede obrazovky objaví okno s názvom **Connect to Server**. V tomto dialógovom okne necháme Server type (Database Engine), vložíme **Server name**, typ prihlásenia zmeníme z Windows Authentication na **SQL Server Authentication**, vložíme používateľské meno a heslo. Potom klikneme na tlačidlo **Connect**.

Následne v ľavej časti obrazovky uvidíme **Object Explorer**, kde je naša pripojená databáza. V tejto databáze vieme prezerať tabuľky, dáta a vykonávať operácie. Používateľské rozhranie nástroja je znovu podobné SQL Developerovi a iným. V **Object Explorer** klikneme pravým tlačidlom na databázu, následne prejdeme do **Tasks -> Export Data** ale môžeme kliknúť aj na Import dáta. Sprievodca je rovnaký. Ukážku vidíme na Obr. 15.



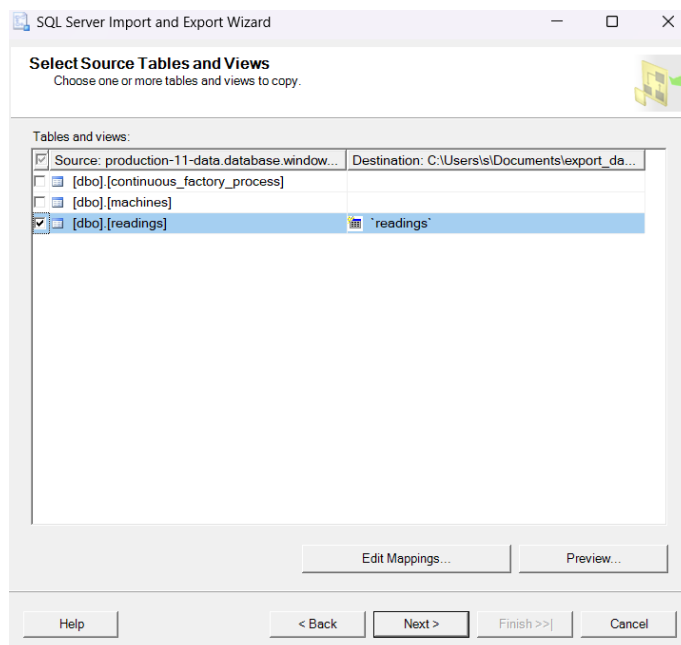
Obr. 15 SSMS spustenie exportu dát

Otvorí sa nám Dialógové okno. Ako prvé uvidíme uvítaciu správu, klikneme na **Next**. V časti Data Source vyberieme SQL Server Native Client 11.0. Následne vyplníme údaje k našej databáze, z ktorej budeme exportovať. Tento sprievodca je univerzálny pre import a aj export. To znamená, že pri importe dát by sme ako Data Source mohli vybrať napr. MS Excel, csv alebo pripojenie k inej databáze.



Obr. 16 SSMS sprievodca – výber zdroja

Po kliknutí na **Next** si potrebujeme vybrať cieľ, kde chceme údaje exportovať a v akom formáte. Pre ukážku zvolíme Microsoft Excel, verziu 97-2003. Následne po kliknutí na **Next**, vyberieme ako sa Obr. 17 budú dáta spracovávať. Vieme vybrať všetky dáta alebo použiť vlastné dopyty pre výber dát. Nasledne vyberáme databázové tabuľky, ktoré chceme exportovať. Cez tlačidlo Preview vieme skontrolovať, ako sprievodca vidí dáta a ako ich exportuje.



Obr. 17 SSMS sprievodca – výber entít

Posledný krok v sprievodcovi je výber akcie. Chceme tieto dáta exportovať hneď alebo uložiť proces ako projekt, ktorý bude možné spustiť neskôr. Vyberieme možnosť spustiť ihneď. Kliknutím na **Finish** sa spustí proces exportu dát. Ak sa vyskytne chyba uvidíme v ktorom kroku sa tak stalo. Zároveň je vygenerovaný aj log s touto chybou. Ak sa všetko podarilo, uvidíme zelené kvačky. Následne tieto dáta importujeme do Oracle databázy pomocou nástroja Oracle SQL Developer. Proces importu csv resp. xls je rovnaký ako v prvom príklade.

1.6.5. Príklad 5 – MSSQL – vlastný skript

Ak preferujeme úplnú kontrolu nad dátami alebo potrebujeme vytvoriť skripty podľa špecifických požiadaviek, môžeme použiť napr. programovací jazyk Python na vytvorenie vlastnej sady nástrojov. V príklade uvedenom nižšie budeme pomocou jazyka Python exportovať dáta z Microsoft SQL databázy, uložíme ich ako Pandas Dataframe a nakoniec ich vložíme do databázy Oracle. Vieme si program prispôbiť podľa seba, nastaviť vlastné parametre pre transformáciu dát, nastaviť periodické aktualizovanie dát a pod. Najlepšie je použiť nástroj Instant Client od Oracle, ale je možné použiť aj JDBC konektory. Pre nástroj Instant Client je potrebné nastaviť premenné prostredia. Na Linuxových distribúciách môže byť tento proces komplikovanejší. Ak zvládneme

tento proces, dokážeme nasadiť vlastný program na ľubovoľný operačný systém. Výhodou je, že pomocou vstavaných knižníc a konektorov, vieme skript prispôbiť aj iným databázam. Nezáleží na použítom operačnom systéme, skript bude zvyčajne spustený v intranete podniku. Linuxové distribúcie poskytnú lepšiu bezpečnosť a lepšie možnosti administrácie.

Rozhodli sme sa použiť knižnicu cx_Oracle a InstantClient na operačnom systéme Windows 10. Z webových stránok spoločnosti Oracle stiahneme Instant Client pre systémy s 64 bit. architektúrou. Rozbalíme knižnicu napr. na disk C. Následne nastavíme premenné prostredia.

Ďalej je potrebné si nastaviť prostredie pre spustenie programov v jazyku Python. Tento postup neopisujeme, nakoľko sa jedná o jednoduchú činnosť. Samotný skript sa skladá z viacerých krokov. Najprv potrebujeme importovať knižnice, pripojiť sa k databáze a potom získať dáta. Nižšie vidíme príklad skriptu na nastavenie pripojenia k Microsoft SQL Databáze.

```
import pandas as pd
import sqlalchemy as sa

server = 'production-11-data.database.windows.net'
database = 'production11'
username = 'production'
password = 'secretpassword'

cnxn_params = f'DRIVER={{SQL Server Native Client
11.0}};SERVER=tcp:{server};DATABASE={database};ENCRYPT=yes;UID={username};PWD={password}'
cnxn_string = f'mssql+pyodbc:///?odbc_connect={cnxn_params}'
engine = sa.create_engine(cnxn_string,echo=True)
```

Skript 3 – Ukážka definovania pripojenia k MSSQL DBMS

Následne si nadefinujeme SQL dopyty, ktoré sa spustia.

```
get_query = 'SELECT * FROM dbo.readings'
```

Skript 4 – Príklad definície dopytu

Proces získavania dát sa vykoná napr. v definovanej funkcii. Máme mnoho možností, môžeme uložiť dáta ako pandas DataFrame, Pickle, Parquet alebo priamo do súboru typu csv.

```
def get_from_table(engine, sql_query, chunksize=1000, csv_file_path="./export.csv",
header=True):
    dfs = []
    df = pd.DataFrame()
    with engine.connect().execution_options(stream_results=True,
max_row_buffer=chunksize) as connection:
        mode = "w"
        for chunk in pd.read_sql(sql_query, connection, chunksize=chunksize):
            dfs.append(chunk)
    df = pd.concat(dfs, ignore_index=True)
    df.to_csv(csv_file_path, mode=mode, header=header, index=False)
```

```
return df
```

Skript 5 – Príklad získania a uloženia dát do DataFrame

Použitím *with* zabezpečíme správne uzatvorenie spojenia. Samotným spúšťačom dopytu je funkcia od Pandas *read_sql*. Dáta získavame po častiach. Ak by sme tak nespravili, proces by bol zdĺhavejší a náročnejší na operačnú pamäť.

```
readings_table = get_from_table(engine, get_query)
```

Skript 6 – Spustenie dopytu

Dataframe si pomenujeme premennou napr. *readings_table*. V ďalšej časti programu definujeme pripojenie k Oracle databáze. Podobne ako pri MS SQL vytvárame Connection String.

```
import cx_Oracle

host='147.232.25.197'
port='1521'
sid='mis4'
user='MIS_user'
password='secretpassword'
dsn = cx_Oracle.makedsn(host, port, sid=sid)
oracle_string = f'oracle://{user}:{password}@{dsn}'
engine = sa.create_engine( oracle_string, convert_unicode=False, pool_recycle=10,
pool_size=50, echo=True )
```

Skript 7 – Ukážka definovania pripojenia k DBMS Oracle

Pri transformácií a úprave dát máme možnosť použiť funkciu *to_sql* alebo *to_csv* od Pandas. Niekedy je potrebné pretypovať dáta.

```
object_columns = [c for c in readings_table.columns[readings_table.dtypes ==
'float'].tolist()]
dtyp = {c:sa.types.NUMERIC(7,2) for c in object_columns}

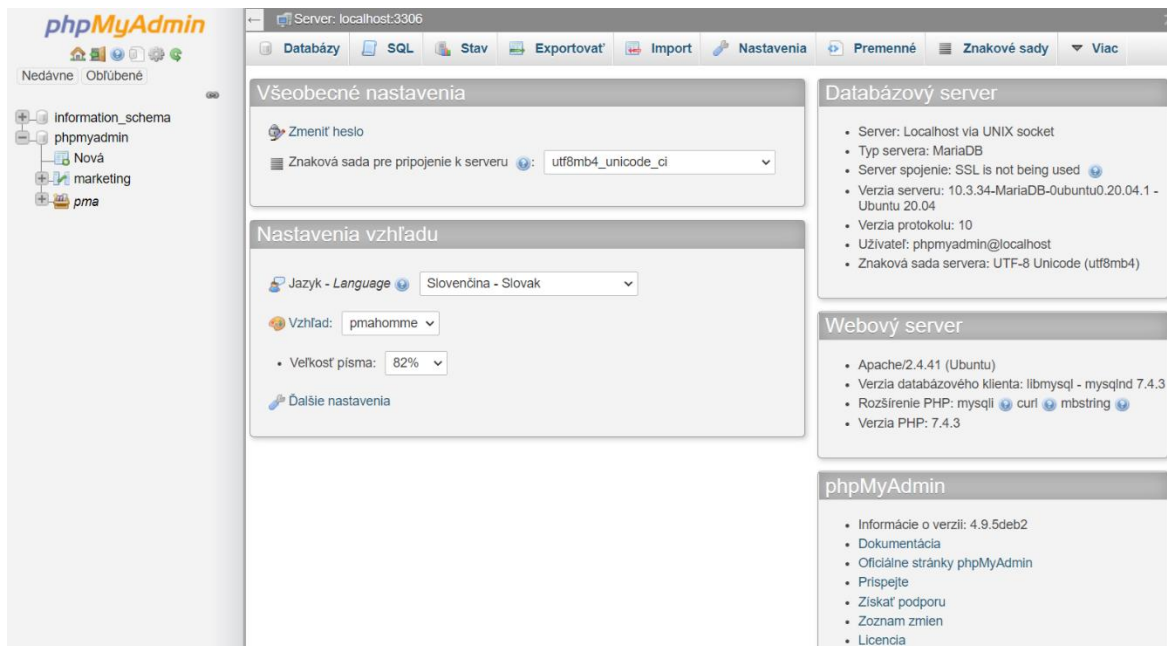
with engine.connect().execution_options(autocommit=True) as connection:
    readings_table.to_sql('machines', con=connection, if_exists='append', chunksize=1000,
index= False, dtype=dtyp)
```

Skript 8 – Príklad pretypovania dát

1.6.6. Príklad 6 – MariaDB – export cez phpMyAdmin

Na ukážku importu dát do databázy Oracle sme vybrali systém phpMyAdmin, ktorý je prepojený s databázou MariaDB. Táto konfigurácia je typická pre informačné systémy používajúce webové rozhrania a PHP. Prvým krokom je prihlásiť sa cez webové rozhranie do systému. Po prihlásení sa dostaneme do prostredia nástroja. V tomto prípade nenastavujeme žiadne pripojenia k databázam. Rozdelenie nástroja je podobné ako pri skôr spomínaných nástrojoch. V ľavej časti obrazovky nájdeme databázy a entity v nich. V hornej časti okna máme hlavné menu. Nájdeme tu informácie o databázach, SQL editor, nastavenia a ďalšie užitočné informácie. Hneď po prihlásení vidíme,

aktuálne nastavenia pre danú inštanciu. Nájde tu informácie o samotnom operačnom systéme, webovom prostredí a databáze.



Obr. 18 Úvodná obrazovka phpmyadmin

Najprv sa vyberie tabuľka, resp. databáza v ľavom menu a potom v hornom menu klikneme na **Exportovať**. Zobrazí sa súbor možností, pomocou ktorých nastavíme export dát napr. takto:

- **Prvá možnosť** je výber šablóny exportu dát, nestáva sa často, aby sme našli šablónu vhodnú priamo pre MIS. Budeme teda postupovať manuálne.
- Ďalej výberom vlastnej **metódy exportu** sa zobrazia možnosti ako na **Chyba! Nenašiel sa žiaden zdroj odkazov..**
- V časti **formát** vyberáme typ súboru, na výber máme csv, Excel, OpenDocument, json, xml, yaml, sql a ďalšie.
- V sekcii **tabuľky** vyberieme tie tabuľky, ktoré chceme exportovať.
- V sekcii **výstup** je najdôležitejšie nastavenie Exportovať tabuľky ako samostatné súbory, to nám v budúcnosti uľahčí prácu.
- Pri **špecifických voľbách** si vyberáme oddeľovač, prípadne znak ohraničujúci začiatok a koniec buniek, náhradnú hodnotu pre NULL a hlavičku dokumentu.
- Po kontrole správnosti parametrov pre export môžeme kliknúť na tlačidlo **Vykonaj**.

Ak pri špecifických voľbách necháme úvodzovky ako ohraničenia polí, všetky údaje sa tak prekonvertujú na reťazce. Je lepšie ich odstrániť, aby sme získali očistené dáta. Samotný sprievodca importu dát si vie zmeniť dátový typ na základe atribútu vytvorenej tabuľky v databáze. Csv alebo Excel súbory by sme importovali podľa **príkladu č. 1** v tejto podkapitole.

Server: localhost:3306 » Databáza: phpmyadmin

Štruktúra SQL Hľadať Dopyt podľa príkladu Exportovať Import Operácie Rutiny Viac

Exportujem tabuľky z databázy "phpmyadmin"

Exportné šablóny:

Nová šablóna: Existujúce šablóny: Šablóna: **1.**

Metóda exportu:

Rýchla - zobrazí len minimum možností **2.**
 Vlastná - zobrazí všetky voľby nastavení

Formát:

3.

Tabuľky:

Tabuľky	Štruktúra	Dáta
<input checked="" type="checkbox"/> Vybrať všetko	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> houses	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> marketing	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/> pma_bookmark	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/> pma_central_columns	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/> pma_column_info	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/> pma_designer_settings	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/> pma_export_templates	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
<input type="checkbox"/> pma_favorite	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

4.

Výstup:

Rename exported databases/tables/columns
 Použiť LOCK TABLES príkaz
 Uložiť výstup do súboru

Šablóna pre názov súboru: použiť aj pre budúce exporty **5.**

Znaková sada súboru:

Kompresia:

Exportovať tabuľky ako samostatné súbory

Zobrazí výstup ako text

Skip tables larger than MiB

Formát - špecifické voľby:

Polia oddelené s:

Polia uzatvorené do:

Polia uvedené pomocou:

Riadky ukončené:

Nahraď NULL hodnoty:

Remove carriage return/line feed characters within columns
 Pridať názvy polí na prvý riadok

7.

Konzola

Obr. 19 Nastavenie parametrov exportu dát

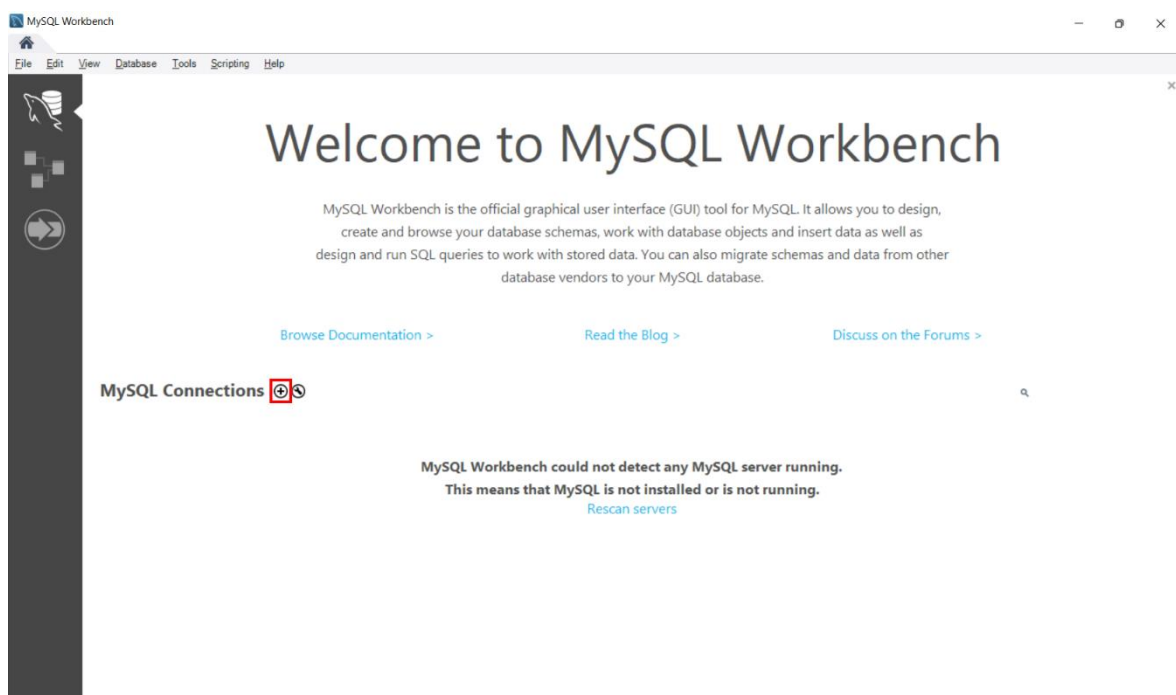
2. Dodatočné informácie

Táto časť obsahuje dodatočné informácie, ktoré sme zhromaždili, použili a zorganizovali, aby boli nápomocné pri replikácií postupov.

2.1. Nastavenie prostredia MySQL Workbench

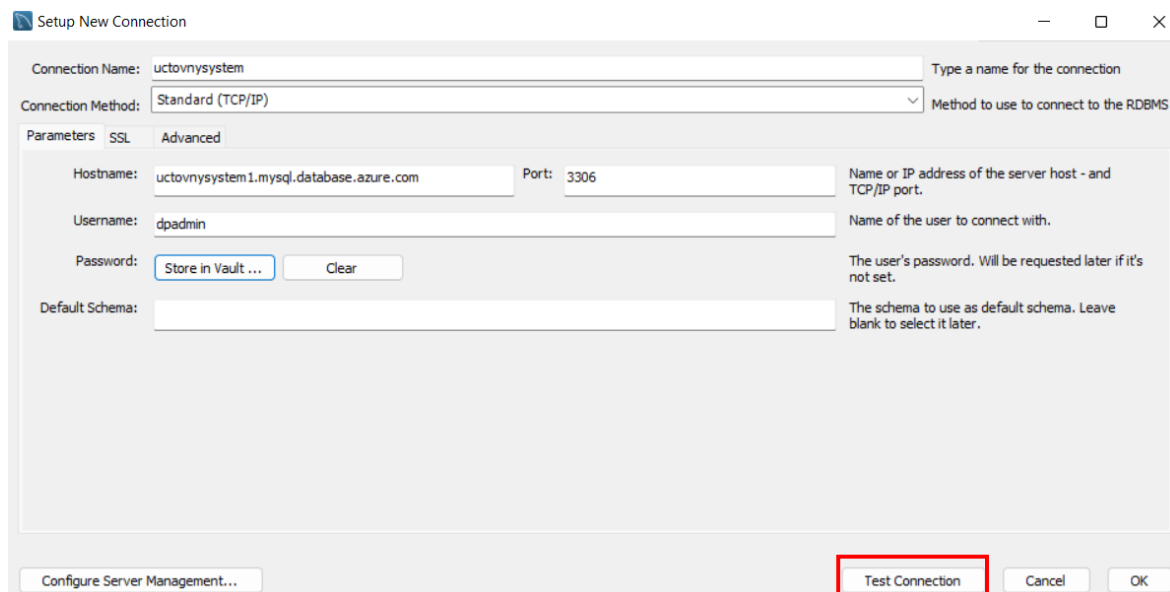
Použijeme program MySQL Workbench na pripojenie sa k MySQL databáze. Ten si stiahneme z webu dev.mysql.com. Vyberieme si inštalačný súbor podľa našich potrieb. V tomto návode použijeme MS Windows. Po stiahnutí si program nainštalujeme a spustíme.

Po spustení programu klikneme na symbol + pre pridanie spojenia ako na Obr. 20.



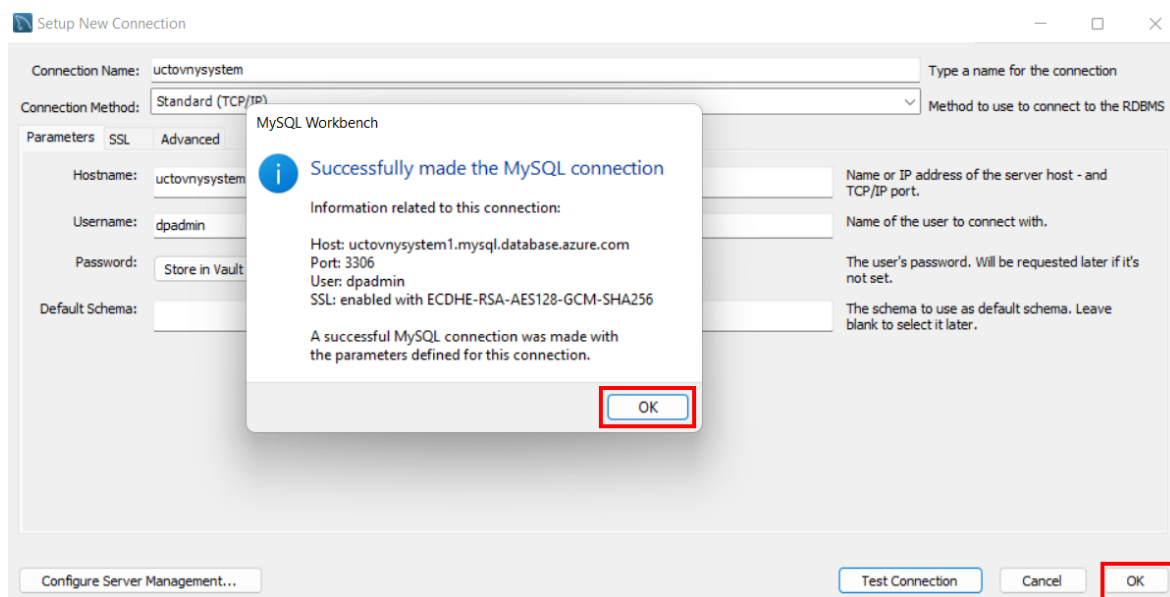
Obr. 20 MySQL Workbench – pridanie spojenia k databáze

Následne sa otvorí dialógové okno kde vyplníme potrebné údaje k pripojeniu. Connection method použijeme napr. TCP/IP, vložíme hostname alebo IP adresu, port, používateľské meno, heslo a prípadne aj samotnú schému (názov databázy) ak zdroj obsahuje viacero databáz.



Obr. 21 MySQL Workbench – sprievodca pridania spojenia k databáze

Po kliknutí na Test Connection zistíme, či je naše nastavenie správne. Zobrazí sa výsledok testu ako napríklad na obrázku nižšie.



Obr. 22 MySQL – správa o úspešnom teste spojenia

Klikneme na Ok a znovu na Ok. Automaticky budeme pripojení k databáze. Prostredie sa podobá Oracle SQL Developerovi.

2.2. Konektory tretích strán v programe Oracle SQL Developer

Pre migráciu databázy MySQL databázy do Oracle je potrebné získať a nastaviť ovládač pre databázu. Postup ilustrujeme na MySQL, ale podobný postup funguje aj pre iné databázy.

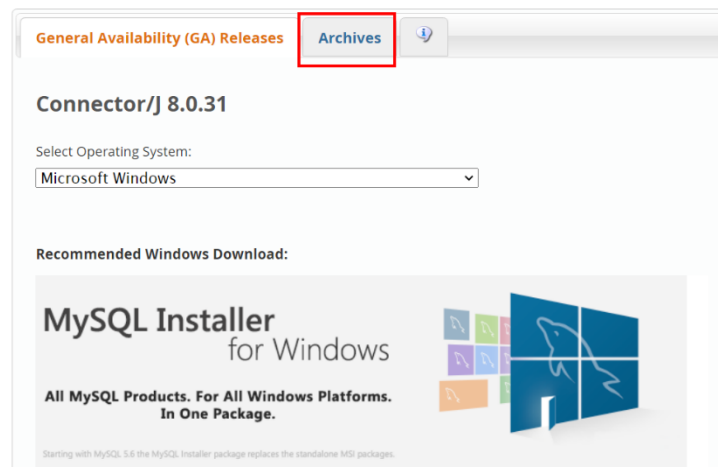
2.2.1. Získanie knižnice

Najprv potrebujeme konektor pre MySQL databázu. Konektory určené pre SQL Developera sú predprogramované knižnice v programovacom jazyku Java. Potrebujeme tzv. JDBC (Java Database Connectivity) knižnicu resp. API, ktoré umožní komunikáciu medzi Oracle a MySQL. SQL Developer od Oracle podporuje tieto knižnice. Táto štandardizovaná knižnica má príponu **jar**. Tvorca databázového systému poskytuje dokumentáciu a zároveň konektory.

Začneme tým, že si stiahneme JDBC konektor zo stránky výrobcu. Na webe <https://www.mysql.com/products/connector/> klikneme na Download.

MySQL Community Downloads

Connector/J



Obr. 23 Web pre stiahnutie ovládačov MySQL

Vyberieme správnu verziu a nezávislosť na platforme. Klikneme na Download napr. zip archívu.

MySQL Product Archives

MySQL Connector/J (Archived Versions)

⚠ Please note that these are old versions. New releases will have recent bug fixes and features!
To download the latest release of MySQL Connector/J, please visit [MySQL Downloads](#).

Product Version:

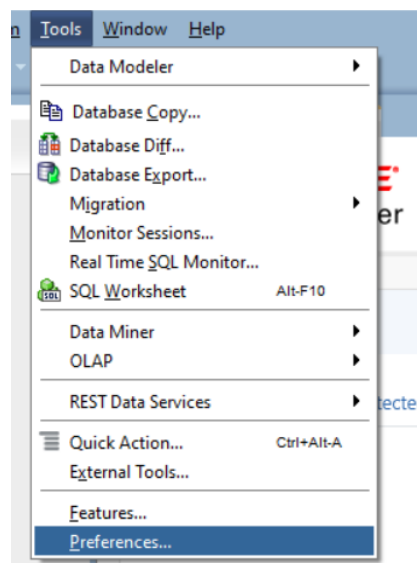
Operating System:

Platform Independent (Architecture Independent), Compressed TAR Archive <small>(mysql-connector-java-8.0.30.tar.gz)</small>	Jul 1, 2022	4.1M	Download
Platform Independent (Architecture Independent), ZIP Archive <small>(mysql-connector-java-8.0.30.zip)</small>	Jul 1, 2022	4.9M	Download

ⓘ We suggest that you use the [MD5 checksums](#) and [GnuPG signatures](#) to verify the integrity of the packages you download.

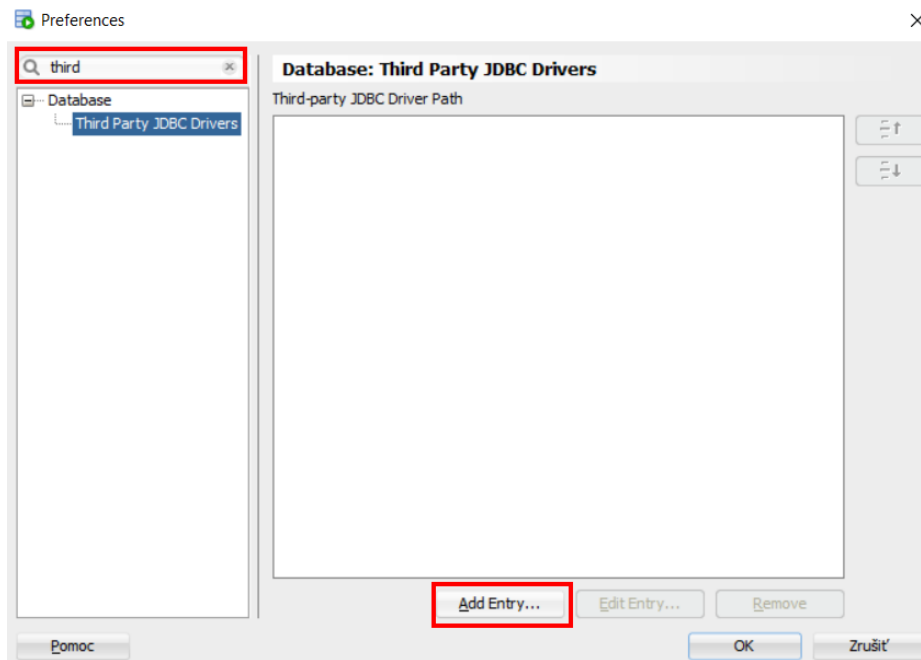
Obr. 24 Výber správnej verzie MySQL ovládača

Balík zip rozbalíme, nájdeme súbor s príponou **jar**, následne ho preniesieme napr. do /Appdata/Roaming/SQL Developer/ (MS Windows). Otvoríme SQL Developer. V menu klikneme na Tools -> Preferences.



Obr. 25 Oracle SQL Developer – otvorenie nastavení

Následne vyhľadáme Third Party JDBC Drivers a klikneme na Add Entry.

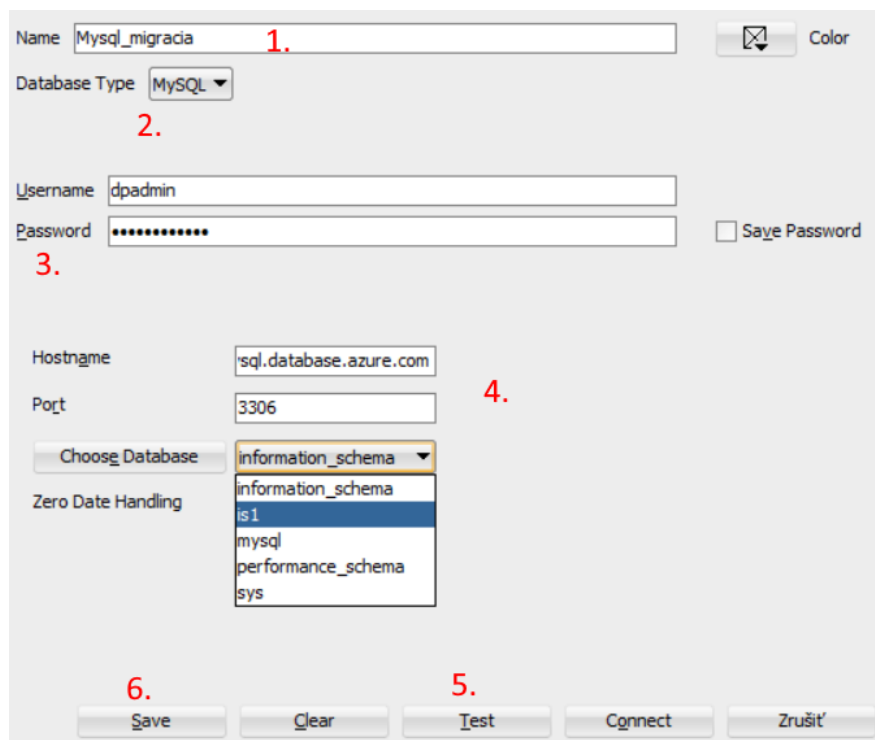


Obr. 26 Oracle SQL Developer – pridanie knižníc tretích strán

Nájdeme na disku konektor a pridáme. Následne klikneme na Ok.

2.2.2. Overenie funkčnosti

V paneli Connections klikneme na New Connection. V dialogovom okne, vložíme názov (1), vyberieme typ MySQL (2), vložíme prihlasovacie údaje (3), vložíme IP alebo hostname a port (4). Po kliknutí na Choose Database sa načítajú uložené databázy, vyberieme tú ktorú budeme migrovať. Klikneme na Test (5) a nakoniec uložíme pripojenie (6).

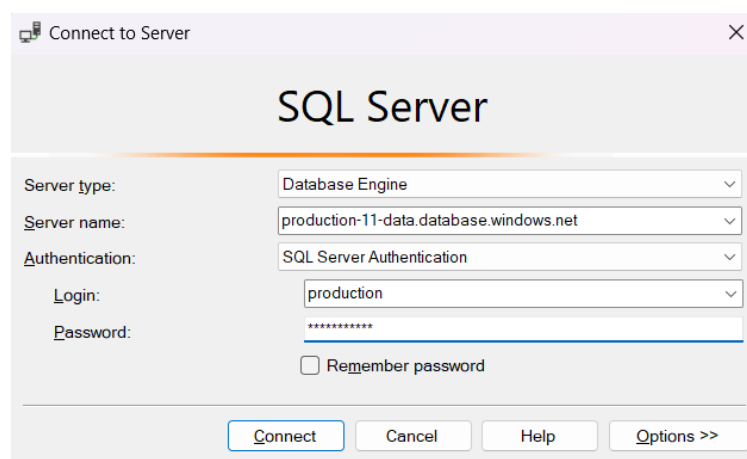


Obr. 27 Oracle SQL Developer – pridanie MySQL spojenia

2.3. Nastavenie prostredia Microsoft SQL

Prvým krokom je získať nástroj a pridať spojenie k databáze. Postup je podobný ako pri MySQL databáze. Stiahneme program SQL Server Management Studio zo stránky výrobcu. Nainštalujeme program. Následne si pridáme pripojenie k našej databáze.

Po spustení programu sa v strede obrazovky objaví okno s názvom Connect to Server.



Obr. 28 SSMS – pridanie spojenia k databáze

V tomto dialógovom okne necháme Server type (Database Engine), vložíme Server name, typ prihlásenia zmeníme z Windows Authentication na SQL Server Authentication, vložíme používateľské meno a heslo. Potom klikneme na tlačidlo Connect.

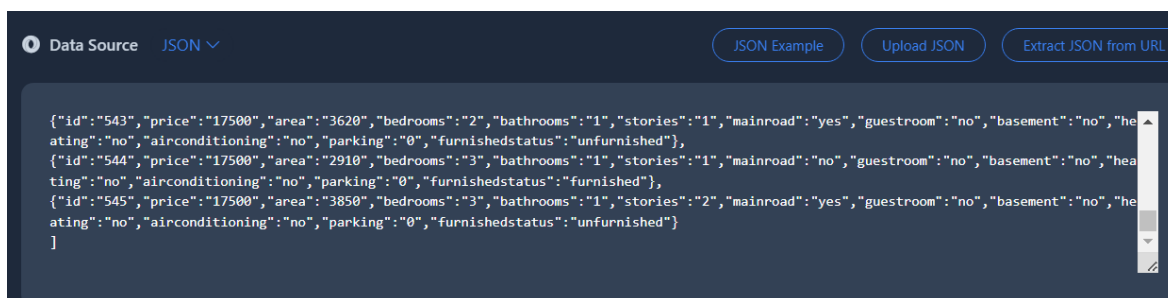
Následne v ľavej časti obrazovky uvidíme Object Explorer, kde je naša pripojená databáza. V tejto databáze vieme prezerať tabuľky, dáta a vykonávať operácie. Používateľské rozhranie nástroja je znovu podobné nástroji Oracle SQL Developer a iným.

2.4. Prevod JSON dát do SQL

Dáta môžu mať rôznu hierarchiu. Tieto dáta je možné spracovávať pomocou vlastných procedúr v samotnom programovacom jazyku SQL. Veľmi často je ale jednoduchšie použiť jazyky ako Python spolu s knižnicami ako Numpy alebo Pandas. Ak nemá používateľ skúsenosti s programovaním, môže použiť webové služby ako napr. <https://tableconvert.com/>.

Tento webový nástroj umožní konvertovať JSON na SQL. Ručne prepísať definície tabuliek nie je zložité. Dôležitejšie je správne prekonvertovať dáta do INSERT INTO príkazov.

Na webe vložíme údaje do editora. Po vložení sa objaví tabuľkové zobrazenie, kde môžeme upravovať dáta. V dolnej časti si vyberieme SQL, vyberiem názov tabuľky, spôsob vkladania dát. Vygenerovaný SQL skript môžeme cez SQL editor spustiť v grafickom nástroji pre manažment databázy.



Obr. 29 Vloženie JSON dát do konvertora

	A	B	C	D	E	F	G
1	id	price	area	bedrooms	bathrooms	stories	mainroad
2	1	133000	7420	4	2	3	yes
3	2	122500	8960	4	4	4	yes
4	3	122500	9960	3	2	2	yes
5	4	122150	7500	4	2	2	yes
6	5	114100	7420	4	1	2	yes
7	6	108500	7500	3	3	1	yes

Obr. 30 Náhľad dát a ich úprava

```
 Generate a create table statement
 Insert multiple rows at once
Table name
predavaneNehnutelnosti
Quotes
No quotes
CREATE TABLE predavaneNehnutelnosti
(
  id varchar(300),
  price varchar(300),
  area varchar(300),
  bedrooms varchar(300),
  bathrooms varchar(300),
  stories varchar(300),
  mainroad varchar(300),
  guestroom varchar(300),
  basement varchar(300),
  heating varchar(300),
  airconditioning varchar(300),
  parking varchar(300),
  furnishedstatus varchar(300)
);
INSERT INTO predavaneNehnutelnosti (id, price, area, bedrooms, bathrooms, stories, mainroad, guestroom,
basement, heating, airconditioning, parking, furnishedstatus) VALUES
('1', '133000', '7420', '4', '2', '3', 'yes', 'no', 'no', 'no', 'yes', '2', 'furnished'),
('2', '122500', '8960', '4', '4', '4', 'yes', 'no', 'no', 'no', 'yes', '3', 'furnished'),
('3', '122500', '9960', '3', '2', '2', 'yes', 'no', 'yes', 'no', 'no', '2', 'semi-furnished'),
('4', '122150', '7500', '4', '2', '2', 'yes', 'no', 'yes', 'no', 'yes', '3', 'furnished'),
('5', '114100', '7420', '4', '1', '2', 'yes', 'yes', 'yes', 'no', 'yes', '2', 'furnished'),
('6', '108500', '7500', '3', '3', '1', 'yes', 'no', 'yes', 'no', 'yes', '2', 'semi-furnished'),
('7', '101500', '8580', '4', '3', '4', 'yes', 'no', 'no', 'no', 'yes', '2', 'semi-furnished'),
('8', '101500', '16200', '5', '3', '2', 'yes', 'no', 'no', 'no', 'no', '0', 'unfurnished');
```

Obr. 31 Ukážka vygenerovaného skriptu s dátami

Záver

Na koniec tejto štúdie spomenieme, že ak by sme dostali údaje v iných formátoch ako napr. JSON, XML, YAML, TXT alebo iné, je potrebné nájsť vhodnú metódu prevodu do podporovaných formátov pre import do Oracle. Samoštúdium tejto problematiky je nevyhnutné. Rozsahovo nie je možné pokryť všetky možné spôsoby konverzie dát. Užívateľ si môže naprogramovať vlastný skript pre potreby ETL procesov, môže použiť online nástroje a pod.

Pre vytvorenie a spustenie databáz sme použili cloudové služby Microsoft Azure. Databázy sme naplnili dátami z fiktívneho podniku. Omnoho dôležitejšie sú postupy než samotné dáta.