



**POLITECHNIKA
RZESZOWSKA**
im. IGNACEGO ŁUKASIEWICZA

Wydział Matematyki i Fizyki Stosowanej
Inżynieria i analiza danych

Projekt
Administracja systemów bazodanowych

Daniel Krzysik 166667

Rzeszów 2022

Spis treści

1. Instalacja maszyny wirtualnej VirtualBox	3
2. Instalacja Windows 10	3
3. Instalacja Oracle	6
4. Projekt.....	10
4.1 Tworzenie kontenerów	10
4.2 Konfiguracja plików listener oraz tnsnames	12
4.3 Utworzenie przestrzeni tabel i użytkowników	17
4.3.1 Użytkownicy wspólni	17
4.3.2 Przestrzeń tabel	20
4.3.3 Użytkownicy lokalni	22
4.4 Wczytanie baz danych do kontenerów.....	28
4.5 Konfiguracja ustawień Oracle's National Language Support (NLS)	37
4.6 Unplug and Plug a PDB	45
4.6.1 Na tej samej maszynie.....	45
4.6.2 Na drugiej maszynie wirtualnej	48

1. Instalacja maszyny wirtualnej VirtualBox

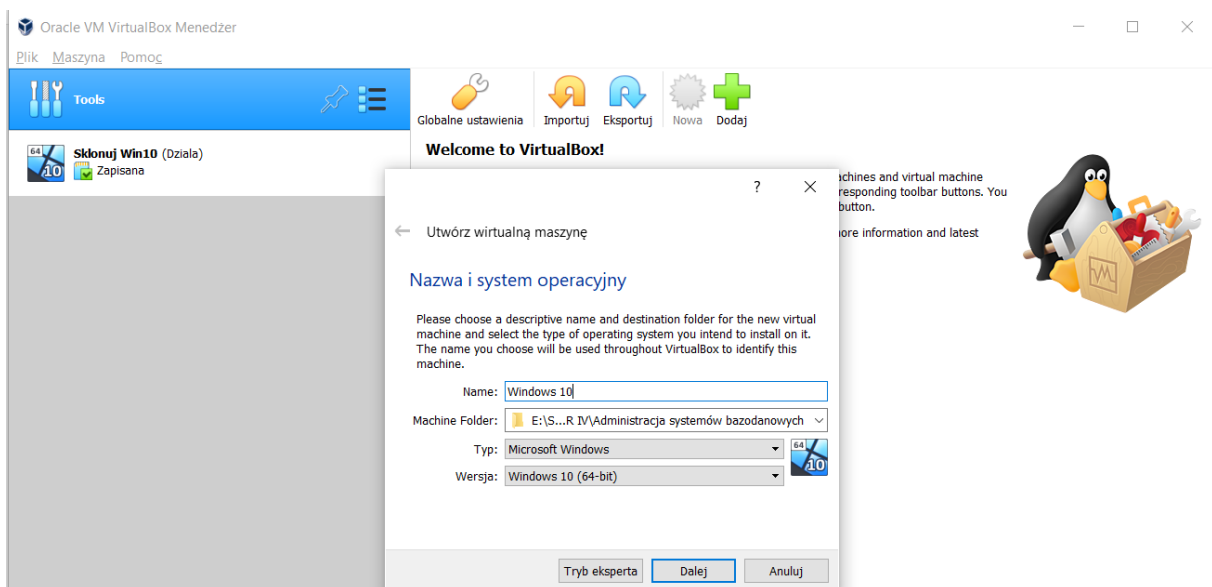
Virtualbox działa na komputerach z systemem Windows, Mac i Linux, więc system Windows 10 można zainstalować na prawie każdej platformie. Pobierz i zainstaluj z linku poniżej.
<https://www.virtualbox.org/wiki/Downloads>

2. Instalacja Windows 10

Zaczynamy od pobrania obrazu systemu Windows 10 (plik iso) z linku poniżej.

<https://www.microsoft.com/pl-pl/software-download/windows10>

Uruchom maszynę wirtualną i kliknij ikonę „Nowa”. Na wyświetlonym ekranie wpisz nazwę swojej maszynie, typ maszyny ustaw na „Microsoft Windows”. Następnie z listy rozwijanej „Wersja” wybierz Windows 10 i wersję 32-bitową lub 64-bitową, w zależności od pobranego ISO. W polu „Machine Folder” wybieramy miejsce docelowe, w którym chcemy aby nasz system został zainstalowany.



Kolejny krok to wybranie ilości pamięci operacyjnej, która zostanie przydzielona danej maszynie wirtualnej po jej uruchomieniu. W przypadku Windowsa 10 zalecane są co najmniej 2 GB, ale warto sięgnąć po więcej. W moim komputerze stacjonarnym mam 16 GB pamięci RAM, więc zdecydowałem, że dla mojej konfiguracji wystarczy 8 GB. Niezależnie od tego, co zdecydujesz, upewnij się, że nie przekroczysz limitu. Jeśli przydzielisz zbyt dużo pamięci RAM, możesz mieć poważne problemy z wydajnością.

← Utwórz wirtualną maszynę

Rozmiar pamięci

Wybierz ilość pamięci (RAM) w megabajtach, która zostanie przydzielona dla wirtualnej maszyny.

Zalecany rozmiar pamięci to: **2048 MB**.



Dalej

Anuluj

Aby zainstalować system Windows 10, należy utworzyć wirtualny dysk twardy do jego instalacji. Dlatego na ekranie, który pojawi się po wybraniu ilości pamięci RAM, kliknij "Stwórz wirtualny dysk twardy".

← Utwórz wirtualną maszynę



Dysk twardy

Jeśli chcesz, to możesz dodać wirtualny dysk twardy do nowej maszyny. Możesz zarówno utworzyć nowy plik twardego dysku jak i wybrać jeden z listy lub z innej lokalizacji, używając ikony folderu.

Jeśli potrzebujesz bardziej złożonej konfiguracji pamięci, to możesz pominąć ten krok i dokonać zmiany ustawień po utworzeniu maszyny.

Zalecana wielkość pliku dysku twardego to: **50,00 GB**.

- ☐ Nie dodawaj wirtualnego dysku twardego
- ☒ Stwórz wirtualny dysk twardy
- ☐ Użyj istniejącego pliku wirtualnego dysku twardego

 Sklonuj Win10.vdi (Normalny, 50,00 GB) 

Utwórz

Anuluj

Jako typ pliku wybierz VDI (VirtualBox Disk Image).

? ×

← Stwórz wirtualny dysk twardy

Typ pliku z wirtualnym dyskiem

Wybierz typ pliku, którego chciałbyś użyć z nowym wirtualnym dyskiem twardym. Jeśli nie potrzebujesz użyć go z innym oprogramowaniem wirtualizacyjnym, to możesz pozostawić to ustawienie niezmienione.

- ☒ VDI (VirtualBox Disk Image)
- ☐ VHD (Virtual Hard Disk)
- ☐ VMDK (Virtual Machine Disk)

Tryb eksperta

Dalej

Anuluj

Dynamiczny przydział (w przeciwieństwie do pamięci operacyjnej) sprawi, że na fizycznym dysku zostanie zajęte tylko tyle miejsca, ile faktycznie jest w danej chwili wykorzystywane.

? ×

← Stwórz wirtualny dysk twardy

Pamięć na fizycznym dysku twardym

Wybierz czy plik nowego wirtualnego dysku powinien rosnąć wraz z użyciem (dynamicznie przydzielany) czy powinien zostać utworzony z maksymalnym rozmiarem (stały rozmiar).

Dynamicznie przydzielany plik twardego dysku będzie używał jedynie miejsca na twoim fizycznym twardym dysku, jeśli się zapełni (do maksymalnego **stałego rozmiaru**), chociaż nie skurczy się automatycznie, gdy zostanie zwolnione miejsce.

Tworzenie stałego rozmiaru pliku może trwać dłużej na niektórych systemach, ale często jest szybszy w użyciu.

- ☒ Dynamicznie przydzielany
- ☐ Stały rozmiar

Dalej

Anuluj


Na koniec pozostaje jeszcze wybór maksymalnego rozmiaru pliku, który symuluje dysk dla maszyny wirtualnej. O ile warunki na to pozwalają, sugerowane 50 GB.

? X


← Stwórz wirtualny dysk twardy

Lokalizacja pliku i rozmiar

Wpisz nazwę dla pliku nowego wirtualnego dysku twardego w polu poniżej lub kliknij na ikonę folderu, by wybrać inny, aby utworzyć w nim plik.

UDIA\SEMESTR IV\Administracja systemów bazodanowych\Windows 10\Windows 10.vdi 

Proszę wybrać rozmiar obrazu wirtualnego dysku twardego w megabajtach. Ten rozmiar jest limitem ilości danych, które wirtualna maszyna może pomieścić na twardym dysku.



Na tym etapie wszystko jest gotowe, aby zainstalować Windowsa 10. Wystarczy wybrać stworzoną maszynę wirtualną i ją uruchomić. Kolejne kroki nie będą się już różnić od instalacji Windowsa 10 w tradycyjnym, komputerze.

3. Instalacja Oracle

Na zainstalowanym Windowsie pobieramy Oracle 18c XE, z linku poniżej.


<https://www.oracle.com/database/technologies/xe18c-downloads.html>

Wypakowujemy archiwum zip i uruchamiamy instalację (setup.exe).

Oracle Database 18c Express Edition X

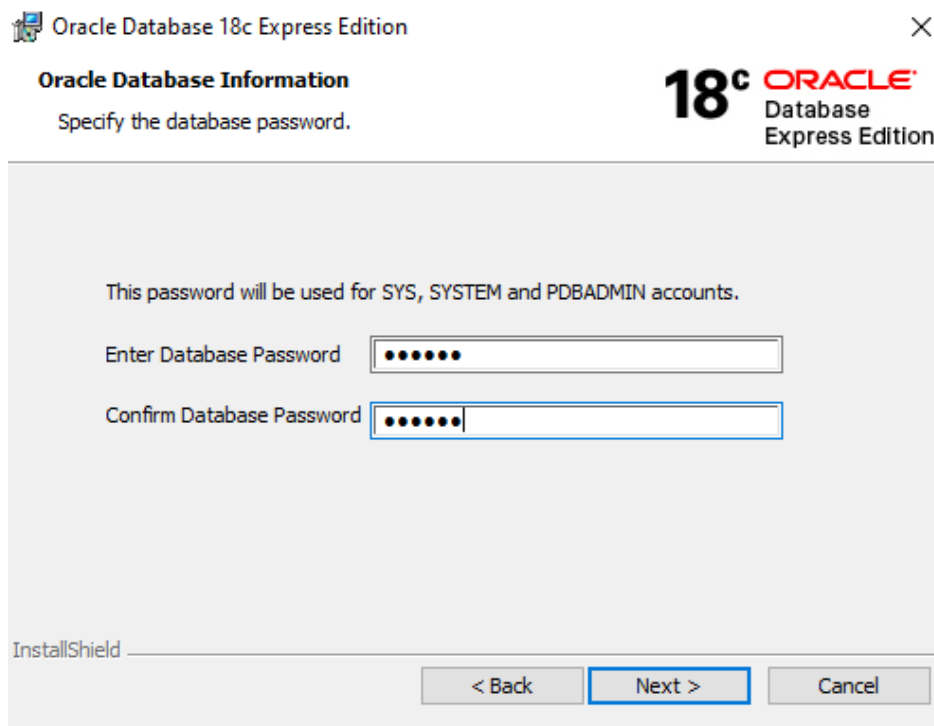
Welcome to the InstallShield Wizard for Oracle Database 18c Express Edition

The InstallShield(R) Wizard will install Oracle Database 18c Express Edition on your computer. To continue, click Next.



Następnie akceptujemy warunki licencji i podajemy lokalizację w której chcemy zainstalować bazę danych i definiujemy hasła dla specjalnych użytkowników systemowych. Warto je sobie dobrze zapamiętać lub zapisać w bezpiecznym miejscu.

Te hasła w przyszłości można zmienić poleceniem ALTER USER <username> IDENTIFIED BY <password>;



Oracle Database 18c Express Edition

Oracle Database Information
Specify the database password.

18^c ORACLE[®]
Database
Express Edition

This password will be used for SYS, SYSTEM and PDBADMIN accounts.

Enter Database Password

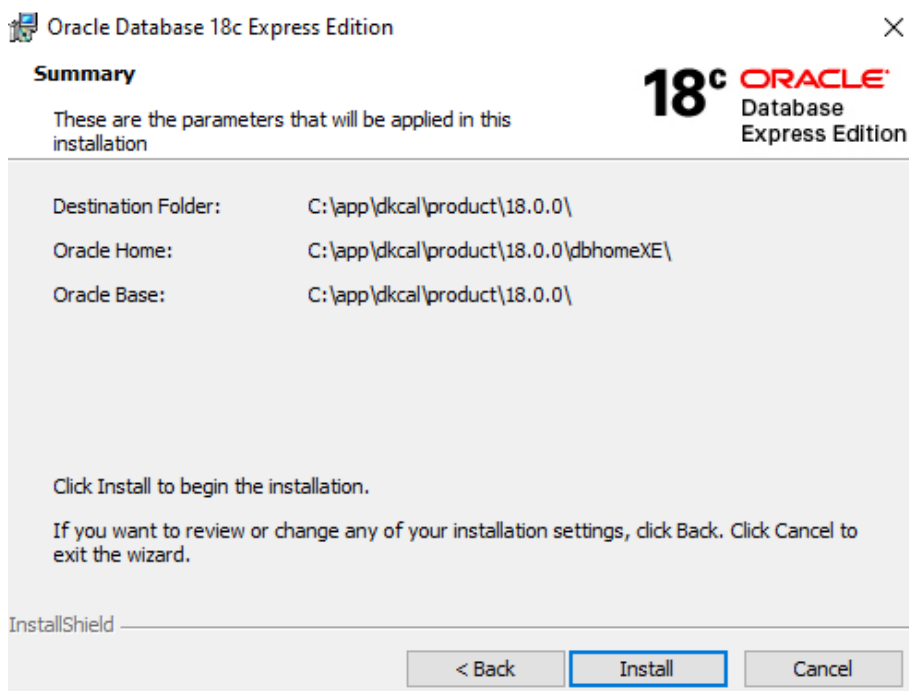
Confirm Database Password

InstallShield

< Back Next > Cancel

Kolejne ważne informacje:

- Oracle Base
- Oracle Home



Oracle Database 18c Express Edition

Summary
These are the parameters that will be applied in this installation

18^c ORACLE[®]
Database
Express Edition

Destination Folder:	C:\app\dkcal\product\18.0.0\
Oracle Home:	C:\app\dkcal\product\18.0.0\dbhomeXE\
Oracle Base:	C:\app\dkcal\product\18.0.0\

Click Install to begin the installation.

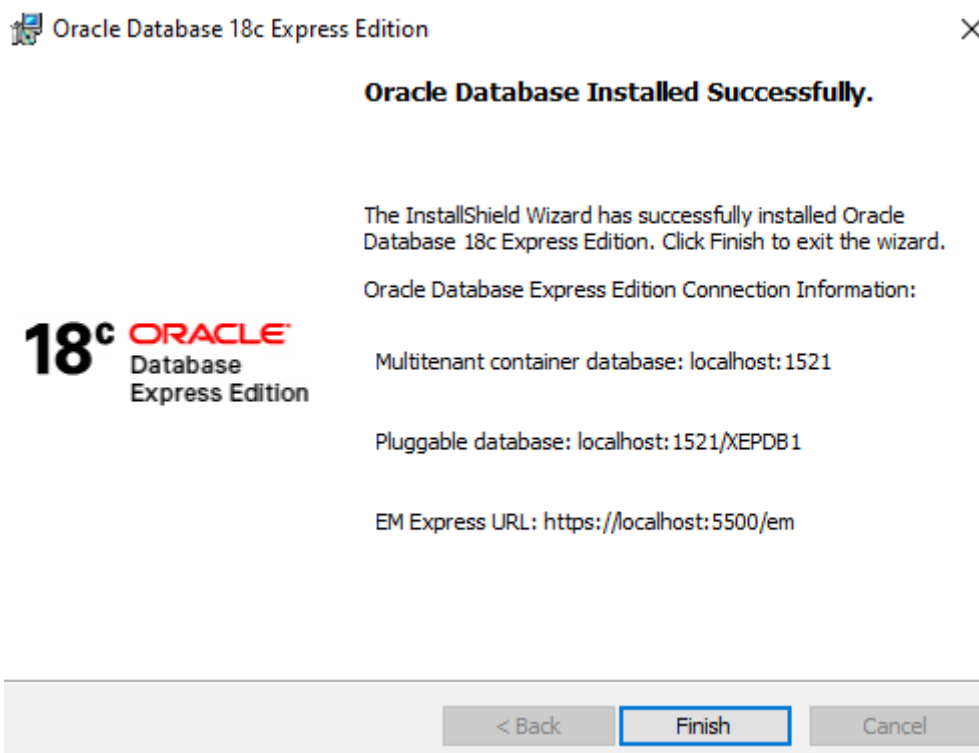
If you want to review or change any of your installation settings, click Back. Click Cancel to exit the wizard.

InstallShield

< Back Install Cancel

W kolejnym kroku zapamiętujemy, bądź notujemy następujące informacje – będą one niezbędne do połączenia się z bazą danych:

- Port: 1521
- Server: localhost
- Database name: XEPDB1
- SID: XE
- Enterprise Express: <https://localhost:5500/em>

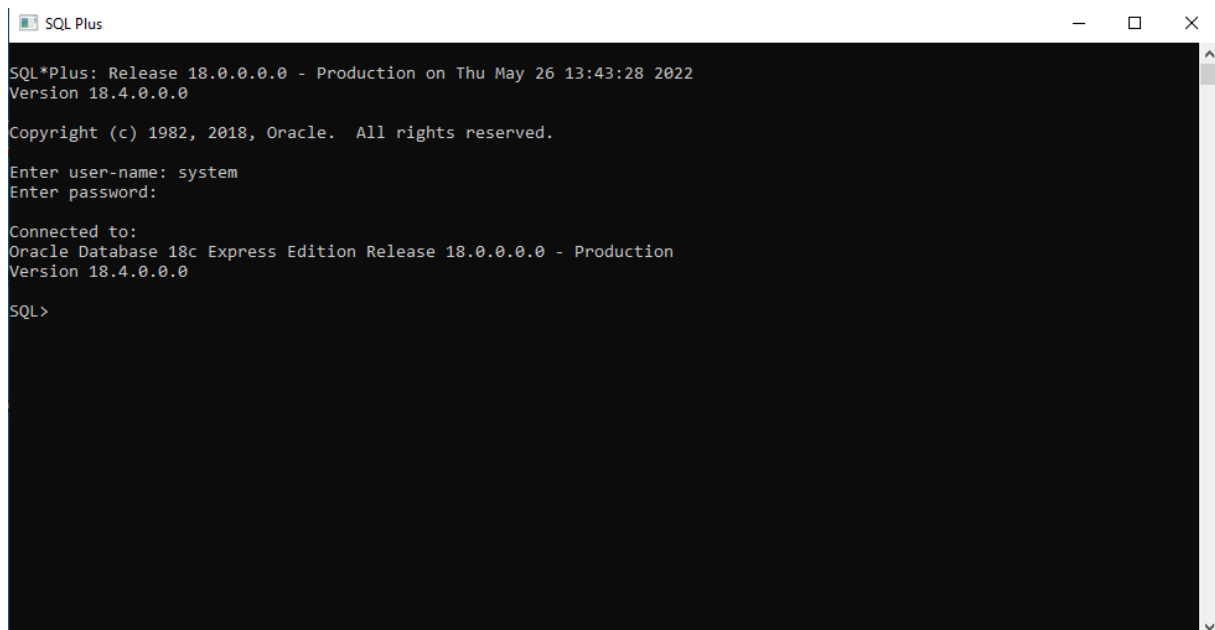


Sprawdźmy połączenie z serwerem.

Oracle Database 18c Express Edition (XE) jest zainstalowana, teraz, aby przetestować jej działanie, połączymy się z serwerem za pomocą standardowego narzędzia konsoli SQL Plus.

W menu Start pojawi się wiele różnych skrótów do uruchamiania programów Oracle Database XE, w tym skrót SQL Plus, uruchamiamy to narzędzie.

Następnie wpisujemy nazwę użytkownika, na przykład SYSTEM i hasło (dokładnie to samo, które podaliśmy podczas procesu instalacji, nie będzie ono wyświetlane podczas wpisywania).



```
SQL*Plus: Release 18.0.0.0.0 - Production on Thu May 26 13:43:28 2022
Version 18.4.0.0.0

Copyright (c) 1982, 2018, Oracle. All rights reserved.

Enter user-name: system
Enter password:

Connected to:
Oracle Database 18c Express Edition Release 18.0.0.0.0 - Production
Version 18.4.0.0.0

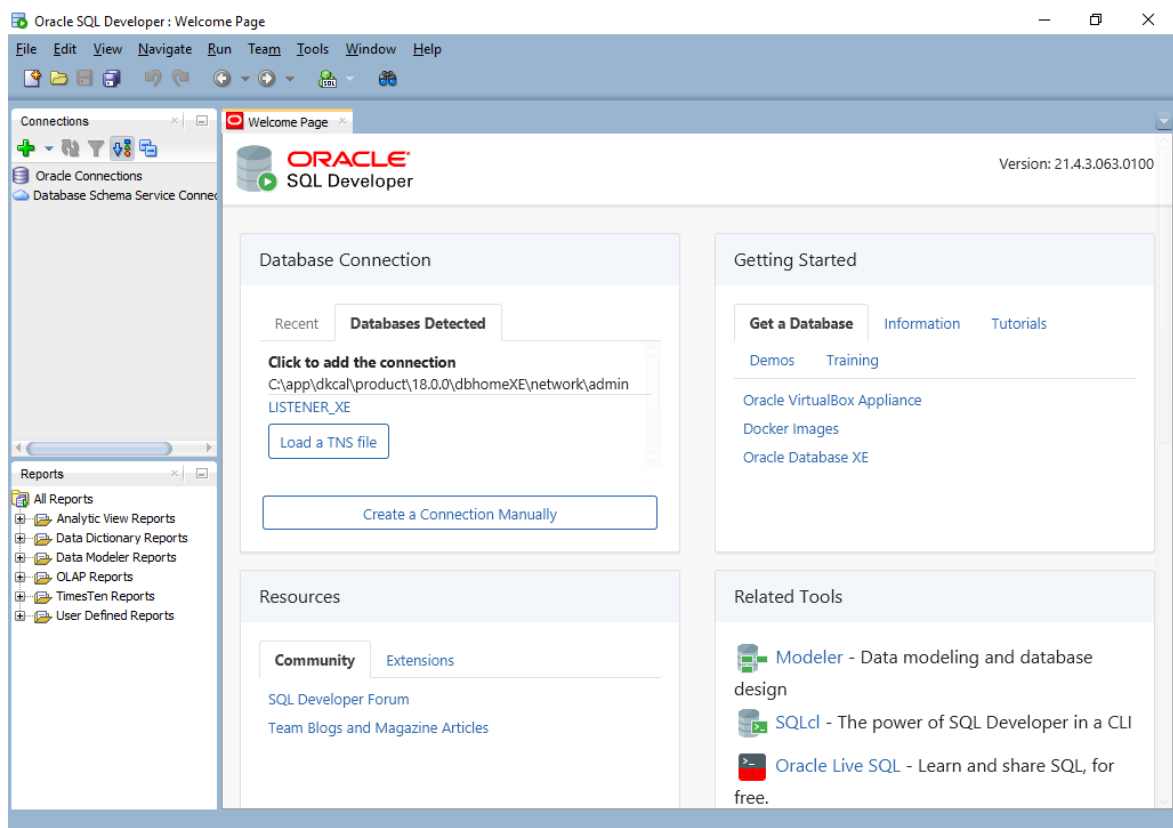
SQL>
```

W ten sposób zainstalowaliśmy bazę danych Oracle Database 18c Express Edition (XE) w systemie operacyjnym Windows 10.

W wykonywaniu naszego projektu przyda nam się jeszcze narzędzie SQL Developer, którego można pobrać z linku poniżej:

[Oracle SQL Developer Downloads](#)

Pobieramy wersję „Windows 64-bit with JDK 8 included”. Następnie plik należy wyodrębnić i wystarczy jedynie uruchomić „sqldeveloper.exe”.



4. Projekt

4.1 Tworzenie kontenerów

Uruchamiamy SQL Plus, a następnie nawiązujemy połączenie z instancją bazy danych.

```
SQL Plus
SQL*Plus: Release 18.0.0.0.0 - Production on Thu May 26 16:33:55 2022
Version 18.4.0.0.0

Copyright (c) 1982, 2018, Oracle. All rights reserved.

Enter user-name: system as sysdba
Enter password:

Connected to:
Oracle Database 18c Express Edition Release 18.0.0.0.0 - Production
Version 18.4.0.0.0
```

Sprawdzamy na jakim kontenerze w tym momencie pracujemy i wyświetlamy informacje o dostępnych kontenerach w bazie danych.

```
SQL> show con_name;

CON_NAME
-----
CDB$ROOT
SQL> show pdbs;

  CON_ID CON_NAME              OPEN MODE  RESTRICTED
-----
      2 PDB$SEED                READ ONLY  NO
      3 XEPDB1                 READ WRITE NO
SQL>
```

Wyświetlamy numeru kontenera, przestrzeń tabel i lokalizację XEPDB1, wykorzystamy te informacje w celu utworzenia kolejnych kontenerów.

```
SQL> SELECT CON_ID, TABLESPACE_NAME, FILE_NAME FROM CDB_DATA_FILES WHERE CON_ID = 3;

  CON_ID TABLESPACE_NAME
-----
      3 SYSTEM
C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB1\SYSTEM01.DBF

      3 SYSAUX
C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB1\SYSAUX01.DBF

      3 UNDOTBS1
C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB1\UNDOTBS01.DBF

  CON_ID TABLESPACE_NAME
-----
      3 USERS
C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB1\USERS01.DBF
```

Usuamy XEPDB1 stworzone podczas instalacji oracle 18c XE, jednakże wcześniej musimy zamknąć bazę.

```
SQL> alter pluggable database XEPDB1 close;
Pluggable database altered.

SQL> DROP PLUGGABLE DATABASE XEPDB1 INCLUDING DATAFILES;
Pluggable database dropped.

SQL> show pdbs;

  CON_ID CON_NAME              OPEN MODE RESTRICTED
-----
      2 PDB$SEED                READ ONLY NO
SQL>
```

Utworzenie kontenerów o nazwach XEPDB1, XEPDB2 oraz XEPDB3, które zostaną stworzone na podstawie kontenera PDB\$SEED.

```
SQL> CREATE PLUGGABLE DATABASE XEPDB1 ADMIN USER pdb1_admin IDENTIFIED BY oracle ROLES=(DBA) FILE_NAME_CONVERT=('C:\app\dkcal\product\18.0.0\oradata\XE\pdbseed', 'C:\app\dkcal\product\18.0.0\oradata\XE\XEPDB1');
Pluggable database created.
```

Stworzyliśmy kontener XEPDB1, nazwa administratora, który został stworzony przy tworzeniu kontenera: pdb1_admin z hasłem oracle. rola DBA została przypisana naszemu użytkownikowi.

Analogicznie tworzymy pozostałe kontenery.

```
SQL> CREATE PLUGGABLE DATABASE XEPDB2 ADMIN USER pdb2_admin IDENTIFIED BY oracle ROLES=(DBA) FILE_NAME_CONVERT=('C:\app\dkcal\product\18.0.0\oradata\XE\pdbseed', 'C:\app\dkcal\product\18.0.0\oradata\XE\XEPDB2');
Pluggable database created.
```

Administrator: pdb2_admin z hasłem oracle.

```
SQL> CREATE PLUGGABLE DATABASE XEPDB3 ADMIN USER pdb3_admin IDENTIFIED BY oracle ROLES=(DBA) FILE_NAME_CONVERT=('C:\app\dkcal\product\18.0.0\oradata\XE\pdbseed', 'C:\app\dkcal\product\18.0.0\oradata\XE\XEPDB3');
Pluggable database created.
```

Administrator: pdb3_admin z hasłem oracle.

Wyświetlamy informacje o dostępnych kontenerach w bazie danych.

```
SQL> show pdbs;

  CON_ID CON_NAME              OPEN MODE RESTRICTED
-----
      2 PDB$SEED                READ ONLY NO
      3 XEPDB1                  MOUNTED
      4 XEPDB2                  MOUNTED
      5 XEPDB3                  MOUNTED
SQL>
```

Przełączamy każdy kontener w tryb read write, aby można zmienić sesję.

```
SQL> alter pluggable database XEPDB1 open;
Pluggable database altered.
```

```
SQL> alter pluggable database XEPDB2 open;
Pluggable database altered.
```

```
SQL> alter pluggable database XEPDB3 open;
Pluggable database altered.

SQL> show pdbs;

  CON_ID CON_NAME              OPEN MODE RESTRICTED
-----
      2 PDB$SEED                READ ONLY NO
      3 XEPDB1                  READ WRITE NO
      4 XEPDB2                  READ WRITE NO
      5 XEPDB3                  READ WRITE NO
SQL>
```

Zmieniamy sesję i ustawiamy kontener XEPDB1.

```
SQL> alter session set container = XEPDB1;
Session altered.
SQL> show con_name;
CON_NAME
-----
XEPDB1
```

Wyświetlenie użytkowników i informacje o tym czy są wspólni. Jak widzimy poniżej użytkownik pdb1_admin jest unikalny dla tego kontenera, jest to nasz administrator stworzony podczas tworzenia bazy.

```
SQL> select username, common from dba_users order by 2;
USERNAME
-----
COM
---
PDB1_ADMIN
NO
ORDSYS
YES
```

Sprawdzamy teraz dla kontenerów XEPDB2 oraz XEPDB3.

```
SQL> alter session set container = XEPDB2;
Session altered.
SQL> show con_name;
CON_NAME
-----
XEPDB2
SQL> select username, common from dba_users order by 2;
USERNAME
-----
COM
---
PDB2_ADMIN
NO
ORDSYS
YES
XS$NULL
YES
```

```
SQL> alter session set container = XEPDB3;
Session altered.
SQL> show con_name;
CON_NAME
-----
XEPDB3
SQL> select username, common from dba_users order by 2;
USERNAME
-----
COM
---
PDB3_ADMIN
NO
ORDSYS
YES
```

4.2 Konfiguracja plików listener oraz tnsnames

Odszukaliśmy i skonfigurowaliśmy zawartość plików listener.ora oraz tnsnames.ora. Pliki znajdują się w poniższej lokalizacji.

C:\app\dkcal\product\18.0.0\dbhomeXE\network\admin

Plik konfiguracyjny listener.ora:

```
listener — Notatnik
Plik Edycja Format Widok Pomoc
# listener.ora Network Configuration File: C:\app\dkcal\product\18.0.0\dbhomeXE\NETWORK\ADMIN\listener.ora
# Generated by Oracle configuration tools.

LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))
      (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))
    )
  )

LISTENER_1 =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1000))
  )

LISTENER_2 =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 2000))
  )

LISTENER_3 =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 3000))
  )
```

Dokonałiśmy konfiguracji procesu listenera, tak aby nasłuchiwał dla protokołu TCP na porcie 1521, 1000, 2000 oraz 3000. Proces o nazwie LISTENER nasłuchuje na porcie 1521 (TCP), LISTENER_1 na porcie 1000 (TCP), LISTENER_2 na porcie 2000 (TCP) oraz LISTENER_3 na porcie 3000 (TCP).

Plik konfiguracyjny tnsnames.ora

```
tnsnames — Notatnik
Plik Edycja Format Widok Pomoc
# tnsnames.ora Network Configuration File: C:\app\dkcal\product\18.0.0\dbhomeXE\NETWORK\ADMIN\tnsnames.ora
# Generated by Oracle configuration tools.

LISTENER_XE =
  (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))

LISTENER_XEPDB1 =
  (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1000))

LISTENER_XEPDB2 =
  (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 2000))

LISTENER_XEPDB3 =
  (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 3000))

XE =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = XE)
    )
  )

XEPDB1 =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1000))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = XEPDB1)
    )
  )

XEPDB2 =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 2000))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = XEPDB2)
    )
  )

XEPDB3 =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 3000))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = XEPDB3)
    )
  )
```

Skonfigurowano plik tnsnames.ora tak, aby można się było połączyć przy pomocy aliasów połączeń tzn. nazw zastępczych kolejno: XEPDB1 do portu 1000, XEPDB2 do portu 2000 oraz XEPDB3 do portu 3000.

Uruchomiliśmy ponownie proces listenerów, wpisując w wierszu poleceń (cmd) poniższe polecenia:

Dla listenera

```
C:\Windows\system32>lsnrctl stop listener

LSNRCTL for 64-bit Windows: Version 18.0.0.0.0 - Production on 26-MAY-2022 23:02:19

Copyright (c) 1991, 2018, Oracle. All rights reserved.

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=localhost)(PORT=1521)))
The command completed successfully

C:\Windows\system32>lsnrctl start listener

LSNRCTL for 64-bit Windows: Version 18.0.0.0.0 - Production on 26-MAY-2022 23:02:33

Copyright (c) 1991, 2018, Oracle. All rights reserved.

Starting tnslnsr: please wait...

TNSLSNR for 64-bit Windows: Version 18.0.0.0.0 - Production
System parameter file is C:\app\dkcal\product\18.0.0\dbhomeXE\network\admin\listener.ora
Log messages written to C:\app\dkcal\product\18.0.0\diag\tnslnsr\DESKTOP-KID4M1G\listener\alert\log.xml
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=127.0.0.1)(PORT=1521)))
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(PIPENAME=\\.\pipe\EXTPROC1521ipc)))

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=localhost)(PORT=1521)))
STATUS of the LISTENER
-----
Alias                listener
Version              TNSLSNR for 64-bit Windows: Version 18.0.0.0.0 - Production
Start Date           26-MAY-2022 23:02:37
Uptime                0 days 0 hr. 0 min. 10 sec
Trace Level           off
Security              ON: Local OS Authentication
SNMP                  OFF
Default Service       XE
Listener Parameter File C:\app\dkcal\product\18.0.0\dbhomeXE\network\admin\listener.ora
Listener Log File     C:\app\dkcal\product\18.0.0\diag\tnslnsr\DESKTOP-KID4M1G\listener\alert\log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=127.0.0.1)(PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(PIPENAME=\\.\pipe\EXTPROC1521ipc)))
Services Summary...
Service "CLRExtProc" has 1 instance(s).
  Instance "CLRExtProc", status UNKNOWN, has 1 handler(s) for this service...
The command completed successfully
```

```
C:\Windows\system32>lsnrctl status listener

LSNRCTL for 64-bit Windows: Version 18.0.0.0.0 - Production on 26-MAY-2022 23:02:50

Copyright (c) 1991, 2018, Oracle. All rights reserved.

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=localhost)(PORT=1521)))
STATUS of the LISTENER
-----
Alias                listener
Version              TNSLSNR for 64-bit Windows: Version 18.0.0.0.0 - Production
Start Date           26-MAY-2022 23:02:37
Uptime                0 days 0 hr. 0 min. 15 sec
Trace Level           off
Security              ON: Local OS Authentication
SNMP                  OFF
Default Service       XE
Listener Parameter File C:\app\dkcal\product\18.0.0\dbhomeXE\network\admin\listener.ora
Listener Log File     C:\app\dkcal\product\18.0.0\diag\tnslnsr\DESKTOP-KID4M1G\listener\alert\log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=127.0.0.1)(PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(PIPENAME=\\.\pipe\EXTPROC1521ipc)))
Services Summary...
Service "CLRExtProc" has 1 instance(s).
  Instance "CLRExtProc", status UNKNOWN, has 1 handler(s) for this service...
The command completed successfully
```

Analogicznie dla listener_1, listener_2 oraz listener_3.

Zmieniamy kontener i otwieramy go.

```
SQL> alter session set container=XEPDB1;
Session altered.

SQL> alter pluggable database XEPDB1 open;
```

Następnie ustawiamy local_listener dla XEPDB1, wydajemy następujące polecenie w celu natychmiastowego zarejestrowania usługi i resetujemy pluggable database.

```
SQL> alter system set local_listener='(address=(protocol=tcp)(host=localhost)(port=1000))' scope=spfile;
System altered.

SQL> alter system register;
System altered.

SQL> alter pluggable database XEPDB1 close immediate;
Pluggable database altered.

SQL> alter pluggable database XEPDB1 open;
Pluggable database altered.
```

Ponownie sprawdzamy status listenera_1.

Za pomocą SQL Plusa nawiązaliśmy połączenie do instancji bazy danych dla utworzonych wcześniej usług z przypisanymi portami.

```
SQL> connect system/system@(description=(address=(host=localhost)(protocol=tcp)(port=1000))(connect_data=(SERVICE_NAME=XEPDB1)))
Connected.
SQL> show con_name;

CON_NAME
-----
XEPDB1
```

Listener_2:

```
SQL> alter session set container=XEPDB2;
Session altered.

SQL> alter pluggable database XEPDB2 open;
alter pluggable database XEPDB2 open
*
ERROR at line 1:
ORA-65019: pluggable database XEPDB2 already open

SQL> alter system set local_listener='(address=(protocol=tcp)(host=localhost)(port=2000))' scope=spfile;
System altered.

SQL> alter system register;
System altered.

SQL> alter pluggable database XEPDB2 close immediate;
Pluggable database altered.

SQL> alter pluggable database XEPDB2 open;
Pluggable database altered.

SQL> connect system/system@(description=(address=(host=localhost)(protocol=tcp)(port=2000))(connect_data=(SERVICE_NAME=XEPDB2)))
Connected.
SQL> show con_name;

CON_NAME
-----
XEPDB2
SQL> _
```

Listener_3:

```
SQL> alter session set container = XEPDB3;
Session altered.

SQL> alter pluggable database XEPDB3 open;
alter pluggable database XEPDB3 open
*
ERROR at line 1:
ORA-65019: pluggable database XEPDB3 already open

SQL> alter system set local_listener='(address=(protocol=tcp)(host=localhost)(port=3000))' scope=spfile;
System altered.

SQL> alter system register;
System altered.

SQL> alter pluggable database XEPDB3 close immediate;
Pluggable database altered.

SQL> alter pluggable database XEPDB3 open;
Pluggable database altered.

SQL> connect system/system@(description=(address=(host=localhost)(protocol=tcp)(port=3000))(connect_data=(SERVICE_NAME=XEPDB3)))
Connected.
SQL> show con_name;

CON_NAME
-----
XEPDB3
SQL>
```

Przetestujemy teraz czy możemy się zalogować do kontenera XEPDB2 przy użyciu portu 1000.

```
SQL> connect system/system@(description=(address=(host=localhost)(protocol=tcp)(port=1000))(connect_data=(SERVICE_NAME=XEPDB2)))
ERROR:
ORA-12514: TNS:listener does not currently know of service requested in connect
descriptor

Warning: You are no longer connected to ORACLE.
SQL>
```

Przetestujemy za pomocą SQL Developera czy nawiążemy połączenie dla XEPDB1 z portem 1000.

New / Select Database Connection

Connection Name	Connection Details
XEPDB1	<p>Database Type: Oracle</p> <p>User Info: Proxy User</p> <p>Authentication Type: Default</p> <p>Username: pdb1_admin, Role: default</p> <p>Password: [masked], Save Password: [unchecked]</p> <p>Connection Type: Basic</p> <p>Details: Advanced</p> <p>Hostname: localhost, Port: 1000</p> <p>Service name: XEPDB1 (selected)</p>

Status : Success

Pomoc Save Clear Test Connect Anuluj

Skonfigurowany plik tnsnames.ora umożliwia połączenie przy pomocy aliasów połączeń tzn. nazw zastępczych, kolejno: XEPDB1 dla portu 1000, XEPDB2 dla portu 2000 oraz XEPDB3 do portu 3000.

```
SQL> connect system as sysdba
Enter password:
Connected.
SQL> connect system/system@XEPDB1
Connected.
SQL> show con_name;

CON_NAME
-----
XEPDB1
SQL> connect system/system@XEPDB2
Connected.
SQL> show con_name;

CON_NAME
-----
XEPDB2
SQL> connect system/system@XEPDB3
Connected.
SQL> show con_name;

CON_NAME
-----
XEPDB3
SQL>
```

4.3 Utworzenie przestrzeni tabel i użytkowników

4.3.1 Użytkownicy wspólni

Utworzenie wspólnych użytkowników, musi odbywać się na kontenerze głównym, nazwa użytkownika musi być poprzedzona prefixem c##.

Utworzenie użytkownika c##pawel

```
SQL> connect system/system as sysdba
Connected.
SQL> CREATE USER c##pawel IDENTIFIED BY pawel CONTAINER=ALL;

User created.

SQL> GRANT CREATE SESSION TO c##pawel CONTAINER=ALL;

Grant succeeded.

SQL> connect c##pawel/pawel@XEPDB1;
Connected.
SQL> connect c##pawel/pawel@XEPDB2;
Connected.
SQL> connect c##pawel/pawel@XEPDB3;
Connected.
SQL>
```

Utworzenie użytkownika c##daniel

```

SQL> connect system/system as sysdba
Connected.
SQL> CREATE USER c##daniel IDENTIFIED BY daniel CONTAINER=ALL;

User created.

SQL> GRANT CREATE SESSION TO c##daniel CONTAINER=ALL;

Grant succeeded.

SQL> connect c##daniel/daniel@XEPDB1;
Connected.
SQL> connect c##daniel/daniel@XEPDB2;
Connected.
SQL> connect c##daniel/daniel@XEPDB3;
Connected.
SQL> show con_name;

CON_NAME
-----
XEPDB3
SQL> show user;
USER is "C##DANIEL"
SQL>

```

Utworzenie użytkownika c##danpaw:

```

SQL> connect system/system as sysdba
Connected.
SQL> CREATE USER c##danpaw IDENTIFIED BY danpaw CONTAINER=ALL;

User created.

SQL> GRANT CREATE SESSION TO c##danpaw CONTAINER=ALL;

Grant succeeded.

SQL> connect c##daniel/danpaw@XEPDB1;
ERROR:
ORA-01017: niepoprawna nazwa uzytkownika/haslo; odmowa zalogowania

Warning: You are no longer connected to ORACLE.
SQL> connect c##danpaw/danpaw@XEPDB1;
Connected.
SQL> connect c##danpaw/danpaw@XEPDB2;
Connected.
SQL> connect c##danpaw/danpaw@XEPDB3;
Connected.
SQL> show con_name;

CON_NAME
-----
XEPDB3
SQL> show user;
USER is "C##DANPAW"
SQL>

```

Wyświetlenie stworzonych użytkowników w kontenerze głównym.

```
SQL> show con_name;

CON_NAME
-----
CDB$ROOT
SQL> select username from dba_users where username like 'C##%';

USERNAME
-----
C##PAWEL
C##DANPAW
C##DANIEL
```

Sprawdźmy teraz czy stworzeni użytkownicy wyświetlają się w kontenerach XEPDB.

Kontener XEPDB1:

```
SQL> connect pdb1_admin/oracle@XEPDB1;
Connected.
SQL> show con_name;

CON_NAME
-----
XEPDB1
SQL> select username from dba_users where username like 'C##%';

USERNAME
-----
C##PAWEL
C##DANPAW
C##DANIEL
```

Kontener XEPDB2:

```
SQL> connect pdb2_admin/oracle@XEPDB2;
Connected.
SQL> show con_name;

CON_NAME
-----
XEPDB2
SQL> select username from dba_users where username like 'C##%';

USERNAME
-----
C##PAWEL
C##DANPAW
C##DANIEL
```

Kontener XEPDB3:

```

SQL> connect pdb3_admin/oracle@XEPDB3;
Connected.
SQL> show con_name;

CON_NAME
-----
XEPDB3
SQL> select username from dba_users where username like 'C##%';

USERNAME
-----
C##PAWEL
C##DANPAW
C##DANIEL

```

4.3.2 Przestrzeń tabel

Sprawdzenie przestrzeni tabel dla każdego kontenera.

XEPDB1:

```

SQL> show con_name;

CON_NAME
-----
XEPDB1
SQL> select tablespace_name from dba_tablespaces;

TABLESPACE_NAME
-----
SYSTEM
SYSAUX
UNDOTBS1
TEMP
SQL>

```

XEPDB2:

```

SQL> show con_name;

CON_NAME
-----
XEPDB2
SQL> select tablespace_name from dba_tablespaces;
-
TABLESPACE_NAME
-----
SYSTEM
SYSAUX
UNDOTBS1
TEMP
SQL> -

```

XEPDB3:

```
SQL> show con_name;

CON_NAME
-----
XEPDB3
SQL> select tablespace_name from dba_tablespaces;

TABLESPACE_NAME
-----
SYSTEM
SYSAUX
UNDOTBS1
TEMP
SQL>
```

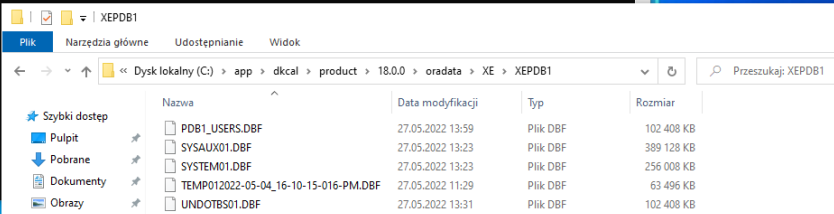
Sprawdzenie lokalizacji i utworzenie przestrzeni tabel dla XEPDB1:

```
SQL> select file_name from dba_data_files;

FILE_NAME
-----
C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB1\UNDOTBS01.DBF
C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB1\SYSAUX01.DBF
C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB1\SYSTEM01.DBF

SQL> CREATE TABLESPACE PDB1_users DATAFILE 'C:\app\dkcal\product\18.0.0\oradata\XE\XEPDB1\PDB1_users.DBF' size 100M autoextend on;
Tablespace created.
SQL> select tablespace_name from dba_tablespaces;

TABLESPACE_NAME
-----
SYSTEM
SYSAUX
UNDOTBS1
TEMP
PDB1_USERS
SQL>
```



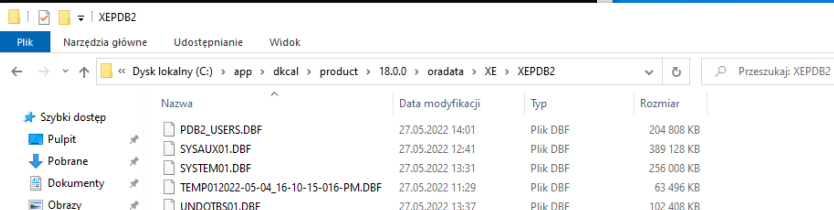
Sprawdzenie lokalizacji i utworzenie przestrzeni tabel dla XEPDB2:

```
SQL> select file_name from dba_data_files;

FILE_NAME
-----
C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB2\UNDOTBS01.DBF
C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB2\SYSAUX01.DBF
C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB2\SYSTEM01.DBF

SQL> CREATE TABLESPACE PDB2_users DATAFILE 'C:\app\dkcal\product\18.0.0\oradata\XE\XEPDB2\PDB2_users.DBF' size 200M autoextend on;
Tablespace created.
SQL> select tablespace_name from dba_tablespaces;

TABLESPACE_NAME
-----
SYSTEM
SYSAUX
UNDOTBS1
TEMP
PDB2_USERS
SQL>
```



Sprawdzenie lokalizacji i utworzenie przestrzeni tabel dla XEPDB3:

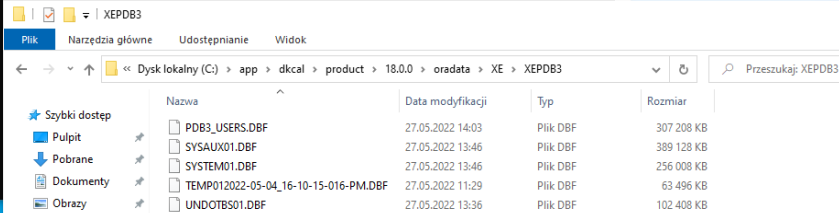
```
SQL> select file_name from dba_data_files;

FILE_NAME
-----
C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB3\UNDOTBS01.DBF
C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB3\SYSAUX01.DBF
C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB3\SYSTEM01.DBF

SQL> CREATE TABLESPACE PDB3_users DATAFILE 'C:\app\dkcal\product\18.0.0\oradata\XE\XEPDB3\PDB3_users.DBF' size 300M autoextend on;
Tablespace created.

SQL> select tablespace_name from dba_tablespaces;

TABLESPACE_NAME
-----
SYSTEM
SYSAUX
UNDOTBS1
TEMP
PDB3_USERS
SQL>
```



4.3.3 Użytkownicy lokalni

Tworzenie użytkowników lokalnych w każdym kontenerze.

Zacznijmy od XEPDB1:

```
SQL> show con_name;

CON_NAME
-----
XEPDB1
SQL> show user;
USER is "PDB1_ADMIN"
SQL> CREATE USER pdb1_daniel identified by daniel;
User created.

SQL> CREATE USER pdb1_pawel identified by pawel;
User created.
```

Teraz utworzymy lokalną rolę, którą nadamy każdemu z użytkowników.

```
SQL> CREATE ROLE pdb1_role;
Role created.

SQL> GRANT CREATE SESSION, CREATE TABLE TO pdb1_role;
Grant succeeded.

SQL> GRANT pdb1_role TO pdb1_daniel;
Grant succeeded.

SQL> GRANT pdb1_role TO pdb1_pawel;
Grant succeeded.
```

Sprawdźmy informacje o naszych użytkownikach.

```
SQL> select username, account_status,created,profile,default_tablespace from dba_users where username LIKE 'PDB1_%';
```

USERNAME	ACCOUNT_STATUS	CREATED
PDB1_PAWEL	OPEN	27-MAY-22
PDB1_ADMIN	OPEN	26-MAY-22
PDB1_DANIEL	OPEN	27-MAY-22

```
SQL>
```

Teraz zmienimy przestrzeń tabel dla naszych użytkowników lokalnych.

```
SQL> alter user PDB1_DANIEL default tablespace PDB1_users;
User altered.
SQL> alter user PDB1_PAWEL default tablespace PDB1_users;
User altered.
SQL> select username, account_status,created,profile,default_tablespace from dba_users where username LIKE 'PDB1_%';
```

USERNAME	ACCOUNT_STATUS	CREATED
PDB1_PAWEL	OPEN	27-MAY-22
PDB1_ADMIN	OPEN	26-MAY-22
PDB1_DANIEL	OPEN	27-MAY-22

Sprawdźmy czy nasi użytkownicy mogą używać polecenia INSERT INTO.

```
SQL> select * from dba_ts_quotas where username LIKE 'PDB1_%';
no rows selected
```

Musimy wprowadzić polecenie:

```
SQL> alter user pdb1_daniel quota unlimited on PDB1_users;
User altered.
SQL> alter user pdb1_pawel quota unlimited on PDB1_users;
User altered.
```

Wyświetlmy raz jeszcze limity przestrzeni tabel dla użytkowników.

```
SQL> select * from dba_ts_quotas where username LIKE 'PDB1_%';
```

TABLESPACE_NAME	USERNAME	BYTES	MAX_BYTES	BLOCKS
PDB1_USERS	PDB1_DANIEL	0	-1	0
PDB1_USERS	PDB1_PAWEL	0	-1	0

Sprawdźmy czy wszystko działa poprawnie:

```

SQL> connect pdb1_daniel/daniel@XEPDB1;
Connected.
SQL> show con_name;

CON_NAME
-----
XEPDB1
SQL> show user;
USER is "PDB1_DANIEL"
SQL> connect pdb1_pawel/pawel@XEPDB1;
Connected.
SQL> show con_name;

CON_NAME
-----
XEPDB1
SQL> show user;
USER is "PDB1_PAWEL"

```

Kolejnym kontenerem jest XEPDB2, działamy w sposób analogiczny:

```

SQL> show con_name;

CON_NAME
-----
XEPDB2
SQL> show user;
USER is "PDB2_ADMIN"
SQL> CREATE USER pdb2_daniel identified by daniel;
User created.

SQL> CREATE USER pdb2_pawel identified by pawel;
User created.

SQL>

```

Teraz utworzymy lokalną rolę, którą nadamy każdemu z użytkowników.

```

SQL> CREATE ROLE pdb2_role;
Role created.

SQL> GRANT CREATE SESSION, CREATE TABLE TO pdb2_role;
Grant succeeded.

SQL> GRANT pdb2_role to pdb2_daniel;
Grant succeeded.

SQL> GRANT pdb2_role to pdb2_pawel;
Grant succeeded.

```

Sprawdźmy informacje o naszych użytkownikach.


```
SQL> select username, account_status, created, profile, default_tablespace from dba_users where username LIKE 'PDB2_%';
```

USERNAME	ACCOUNT_STATUS	CREATED
PROFILE	DEFAULT_TABLESPACE	
PDB2_DANIEL	OPEN	27-MAY-22
DEFAULT	SYSTEM	
PDB2_ADMIN	OPEN	26-MAY-22
DEFAULT	SYSTEM	
PDB2_PAWEL	OPEN	27-MAY-22
DEFAULT	SYSTEM	

Teraz zmienimy przestrzeń tabel dla naszych użytkowników lokalnych.

```
SQL> alter user PDB2_DANIEL default tablespace PDB2_users;
User altered.

SQL> alter user PDB2_PAWEL default tablespace PDB2_users;
User altered.

SQL> select username, account_status, created, profile, default_tablespace from dba_users where username LIKE 'PDB2_%';
```

USERNAME	ACCOUNT_STATUS	CREATED
PROFILE	DEFAULT_TABLESPACE	
PDB2_DANIEL	OPEN	27-MAY-22
DEFAULT	PDB2_USERS	
PDB2_ADMIN	OPEN	26-MAY-22
DEFAULT	SYSTEM	
PDB2_PAWEL	OPEN	27-MAY-22
DEFAULT	PDB2_USERS	

Sprawdźmy czy nasi użytkownicy mogą używać polecenia INSERT INTO.

```
SQL> select * from dba_ts_quotas where username LIKE 'PDB2_%';

no rows selected
```

Musimy wprowadzić polecenie:

```
SQL> select * from dba_ts_quotas where username LIKE 'PDB2_%';

no rows selected

SQL> alter user pdb2_daniel quota unlimited on PDB2_users;
User altered.

SQL> alter user pdb2_pawel quota unlimited on PDB2_users;
User altered.
```

Wyświetlmy raz jeszcze limity przestrzeni tabel dla użytkowników.

```
SQL> select * from dba_ts_quotas where username LIKE 'PDB2_%';
```

TABLESPACE_NAME	USERNAME	BYTES	MAX_BYTES	BLOCKS
MAX_BLOCKS DRO				
PDB2_USERS	PDB2_DANIEL	0	-1	0
-1 NO				
PDB2_USERS	PDB2_PAWEL	0	-1	0
-1 NO				

Sprawdźmy czy wszystko działa poprawnie:

```

SQL> connect pdb2_daniel/daniel@XEPDB2;
Connected.
SQL> show con_name;

CON_NAME
-----
XEPDB2
SQL> show user;
USER is "PDB2_DANIEL"
SQL> connect pdb2_pawel/pawel@XEPDB2;
Connected.
SQL> show con_name;

CON_NAME
-----
XEPDB2
SQL> show user;
USER is "PDB2_PAWEL"

```

Ostatni kontener XEPDB3:

```

SQL> show con_name;

CON_NAME
-----
XEPDB3
SQL> show user;
USER is "PDB3_ADMIN"
SQL> CREATE USER pdb3_daniel identified by daniel;
User created.

SQL> CREATE USER pdb3_pawel identified by pawel;
User created.

SQL> _

```

Teraz utworzymy lokalną rolę, którą nadamy każdemu z użytkowników.

```

SQL> CREATE ROLE pdb3_role;
Role created.

SQL> GRANT CREATE SESSION, CREATE TABLE TO pdb3_role;
Grant succeeded.

SQL> GRANT pdb3_role to pdb3_daniel;
Grant succeeded.

SQL> GRANT pdb3_role to pdb3_pawel;
Grant succeeded.

```

Sprawdźmy informacje o naszych użytkownikach.

```
SQL> select username, account_status, created, profile, default_tablespace from dba_users where username LIKE 'PDB3_%';
```

USERNAME	ACCOUNT_STATUS	CREATED
PDB3_PAWEL	OPEN	27-MAY-22
PDB3_ADMIN	OPEN	26-MAY-22
PDB3_DANIEL	OPEN	27-MAY-22

Teraz zmienimy przestrzeń tabel dla naszych użytkowników lokalnych.

```
SQL> alter user PDB3_DANIEL default tablespace PDB3_users;
User altered.
SQL> alter user PDB3_PAWEL default tablespace PDB3_users;
User altered.
SQL> select username, account_status, created, profile, default_tablespace from dba_users where username LIKE 'PDB3_%';
```

USERNAME	ACCOUNT_STATUS	CREATED
PDB3_PAWEL	OPEN	27-MAY-22
PDB3_ADMIN	OPEN	26-MAY-22
PDB3_DANIEL	OPEN	27-MAY-22

Sprawdźmy czy nasi użytkownicy mogą używać polecenia INSERT INTO.

```
SQL> select * from dba_ts_quotas where username LIKE 'PDB3_%';
no rows selected
```

Musimy wprowadzić polecenie:

```
SQL> alter user pdb3_daniel quota unlimited on PDB3_users;
User altered.
SQL> alter user pdb3_pawel quota unlimited on PDB3_users;
User altered.
```

Wyświetlmy raz jeszcze limity przestrzeni tabel dla użytkowników.

```
SQL> select * from dba_ts_quotas where username LIKE 'PDB3_%';
```

TABLESPACE_NAME	USERNAME	BYTES	MAX_BYTES	BLOCKS
PDB3_USERS	PDB3_DANIEL	0	-1	0
PDB3_USERS	PDB3_PAWEL	0	-1	0

Sprawdźmy czy wszystko działa poprawnie:

```

SQL> connect pdb3_daniel/daniel@XEPDB3;
Connected.
SQL> show con_name;

CON_NAME
-----
XEPDB3
SQL> show user;
USER is "PDB3_DANIEL"
SQL> connect pdb3_pawel/pawel@XEPDB3;
Connected.
SQL> show con_name;

CON_NAME
-----
XEPDB3
SQL> show user;
USER is "PDB3_PAWEŁ"

```

4.4 Wczytanie baz danych do kontenerów

Do kontenera XEPDB1 wczytamy bazę danych komis samochodowy, która została utworzona w poprzednim semestrze. W tym celu użyjemy narzędzia SQL Developer. Zaczniemy od stworzenia nowego połączenia.

New / Select Database Connection

Connection Name: XEPDB1 - komis

Database Type: Oracle

User Info: Proxy User

Authentication Type: Default

Username: pdb1_daniel

Password:

Role: default

Save Password: ☒

Connection Type: Basic

Details: Advanced

Hostname: localhost

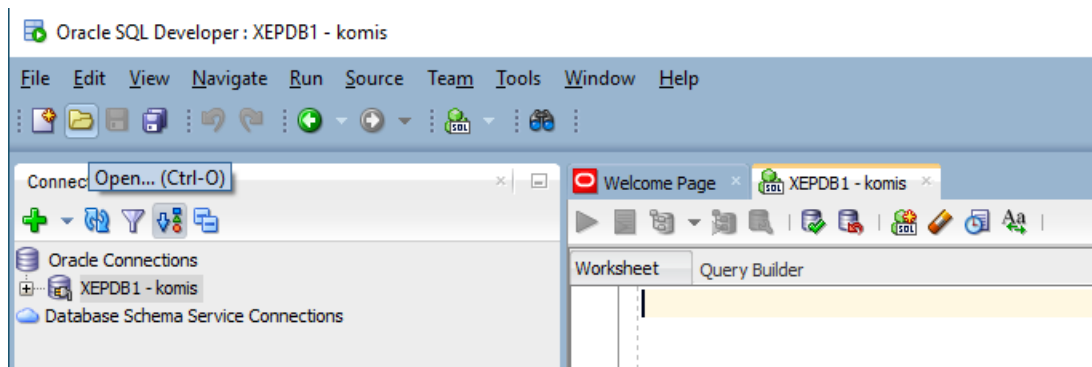
Port: 1000

Service name: XEPDB1

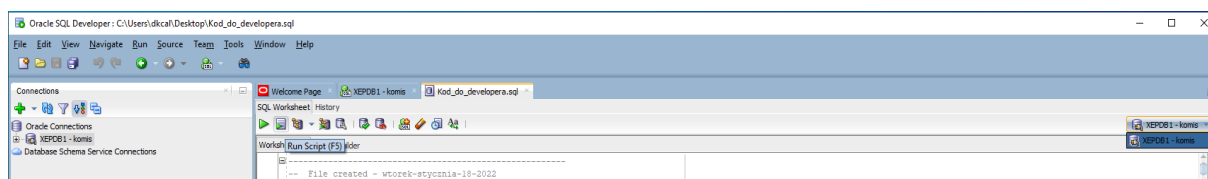
Status: Success

Pomoc Save Clear Test Connect Anuluj

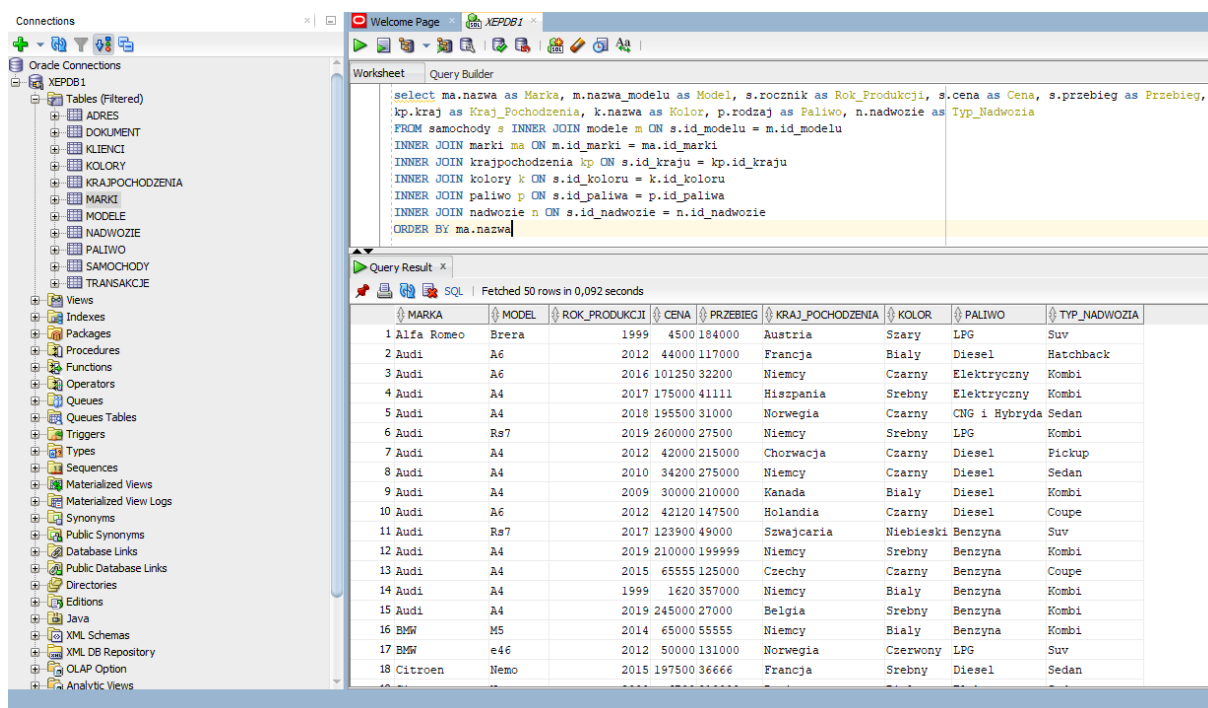
Wczytujemy plik z rozszerzeniem sql, zaznaczamy w lewym górnym rogu „kopertę” i znajdujemy lokalizację naszego pliku.



W prawym górnym rogu wybieramy połączenie, które nas interesuje – w naszym przypadku XEPDB1 – komis. Wczytujemy skrypt F5.



Jak widzimy nasz baza została poprawnie wczytana, dla sprawdzenia wywołaliśmy przykładowe zapytanie.



Wyświetlmy jeszcze przestrzeń tabel dla wczytanej bazy.

select table_name, tablespace_name from user_tables		
Query Result x		
SQL All Rows Fetched: 11 in 0,118 seconds		
	TABLE_NAME	TABLESPACE_NAME
1	ADRES	PDB1_USERS
2	DOKUMENT	PDB1_USERS
3	KLIENCI	PDB1_USERS
4	KOLORY	PDB1_USERS
5	KRAJPOCHODZENIA	PDB1_USERS
6	MARKI	PDB1_USERS
7	MODELE	PDB1_USERS
8	NADWOZIE	PDB1_USERS
9	PALIWO	PDB1_USERS
10	SAMOCHODY	PDB1_USERS
11	TRANSAKCJE	PDB1_USERS

Dla kontenera XEPDB2 odblokujemy bazę Sales History (SH).

Zaczynamy od pobrania schematu ze strony:

[Release Oracle Database 18c Sample Schemas · oracle-samples/db-sample-schemas · GitHub](#)

Wczytanie bazy za pomocą SQL Plus

```
SQL> connect pdb2_admin/oracle@XEPDB2;
Connected.
SQL> show con_name;

CON_NAME
-----
XEPDB2
SQL> show user;
USER is "PDB2_ADMIN"
SQL> @?/demo/schema/sales_history/sh_main.sql sh_pass PDB2_users TEMP oracle C:/app/dkcal/product/18.0.0/dbhomeXE/demo/s
chema/sales_history/ C:/app/dkcal/product/18.0.0/dbhomeXE/demo/schema/log/ v3 localhost:2000/XEPDB2
```

```
PL/SQL procedure successfully completed.
```

```
PL/SQL procedure successfully completed.
```

```
SQL> show user;
USER is "SH"
SQL>
```

Robimy nowe połączenie w SQL Developer jako użytkownik SH.

New / Select Database Connection

Connection Name	Connection Details
XEPDB1	pdb1_pawel@//lo...

Name: XEPDB2

Database Type: Oracle

User Info: Proxy User

Authentication Type: Default

Username: sh

Password:

Role: default

Save Password: ☒

Connection Type: Basic

Details: Advanced

Hostname: localhost

Port: 2000

☐ SID: xe
☒ Service name: XEPDB2

Status: Success

Pomoc Save Clear Test Connect Anuluj

Dla sprawdzenia poprawności wczytania bazy wykonujemy przykładowe zapytanie SQL.

Connections

Oracle Connections

- XEPDB1
- XEPDB2
 - Tables (Filtered)
 - CAL_MONTH_SALES_MV
 - CHANNELS
 - COSTS
 - COUNTRIES
 - CUSTOMERS
 - DR\$SUP_TEXT_IDX\$I
 - DR\$SUP_TEXT_IDX\$K
 - DR\$SUP_TEXT_IDX\$N
 - DR\$SUP_TEXT_IDX\$U
 - FWEEK_PSCAT_SALES_MV
 - PRODUCTS
 - PROMOTIONS
 - SALES
 - SALES_TRANSACTIONS_EXT
 - SUPPLEMENTARY_DEMOGRAPHICS
 - TIMES
 - Views
 - Indexes
 - Packages
 - Procedures
 - Functions
 - Operators
 - Queues
 - Queues Tables
 - Triggers
 - Types
 - Sequences
 - Materialized Views
 - Materialized View Logs
 - Synonyms
 - Public Synonyms
 - Database Links
 - Public Database Links
 - Directories

Welcome Page XEPDB2

Worksheet Query Builder

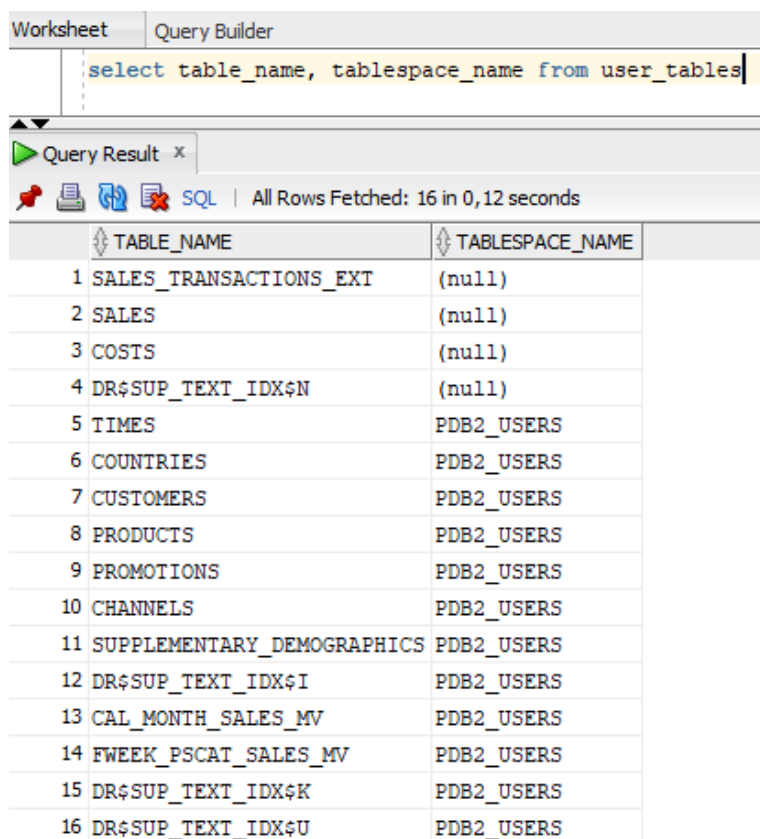
```
SELECT PROD_NAME FROM PRODUCTS
```

Query Result x

SQL | Fetched 50 rows in 0,011 seconds

PROD_NAME
1 SMP Telephoto Digital Camera
2 17" LCD w/built-in HDTV Tuner
3 Envoy 256MB - 40GB
4 Y Box
5 Mini DV Camcorder with 3.5" Swivel LCD
6 Envoy Ambassador
7 Laptop carrying case
8 Home Theatre Package with DVD-Audio/Video Play
9 18" Flat Panel Graphics Monitor
10 Envoy External Keyboard
11 External 101-key keyboard
12 PCMCIA modem/fax 28800 baud
13 SIMM- 8MB PCMCIAII card
14 SIMM- 16MB PCMCIAII card
15 Multimedia speakers- 3" cones
16 Unix/Windows 1-user pack
17 8.3 Minitower Speaker
18 Mouse Pad
19 1.44MB External 3.5" Diskette
20 Multimedia speakers- 5" cones
21 PCMCIA modem/fax 19200 baud
22 External 6X CD-ROM
23 External 8X CD-ROM
24 Envoy External 6X CD-ROM
25 Envoy External 8X CD-ROM

Wyświetlamy przestrzeń tabel dla schematu SH.



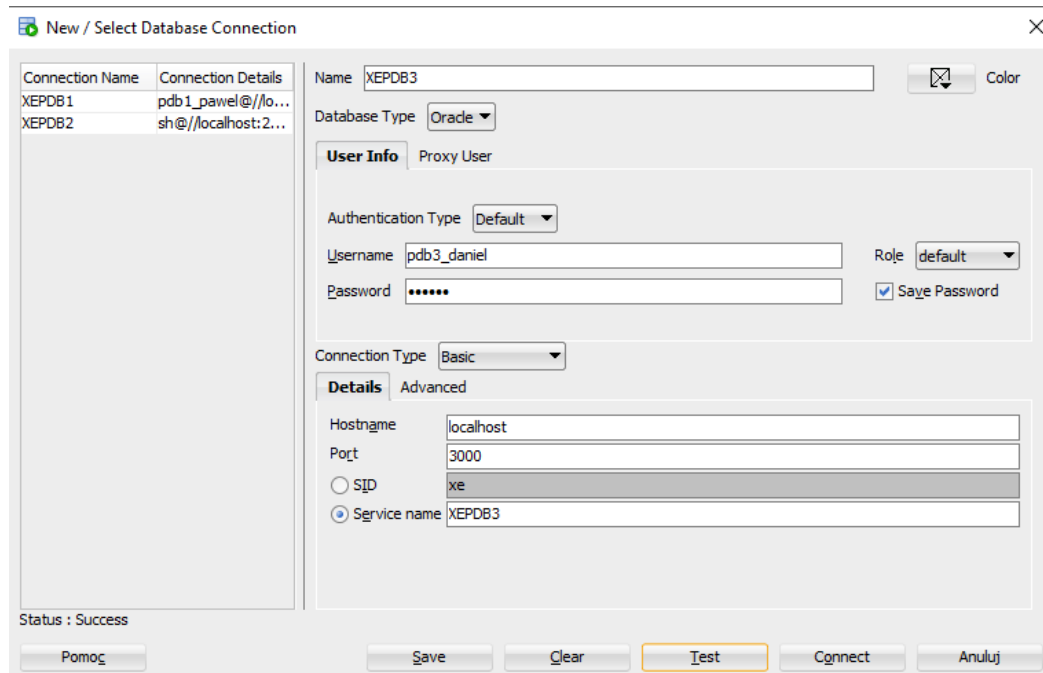
The screenshot shows the SQL Developer interface. At the top, the 'Query Builder' tab is active, displaying the query: `select table_name, tablespace_name from user_tables`. Below the query, the 'Query Result' window shows the results of the query. The status bar indicates 'All Rows Fetched: 16 in 0,12 seconds'. The results are displayed in a table with two columns: 'TABLE_NAME' and 'TABLESPACE_NAME'.

TABLE_NAME	TABLESPACE_NAME
1 SALES_TRANSACTIONS_EXT	(null)
2 SALES	(null)
3 COSTS	(null)
4 DR\$SUP_TEXT_IDX\$N	(null)
5 TIMES	PDB2_USERS
6 COUNTRIES	PDB2_USERS
7 CUSTOMERS	PDB2_USERS
8 PRODUCTS	PDB2_USERS
9 PROMOTIONS	PDB2_USERS
10 CHANNELS	PDB2_USERS
11 SUPPLEMENTARY_DEMOGRAPHICS	PDB2_USERS
12 DR\$SUP_TEXT_IDX\$I	PDB2_USERS
13 CAL_MONTH_SALES_MV	PDB2_USERS
14 FWEEK_PSCAT_SALES_MV	PDB2_USERS
15 DR\$SUP_TEXT_IDX\$K	PDB2_USERS
16 DR\$SUP_TEXT_IDX\$U	PDB2_USERS

Dla kontenera XEPDB3 wczytamy bazę danych Credit Card 5m rekordów, która zawiera ze strony:

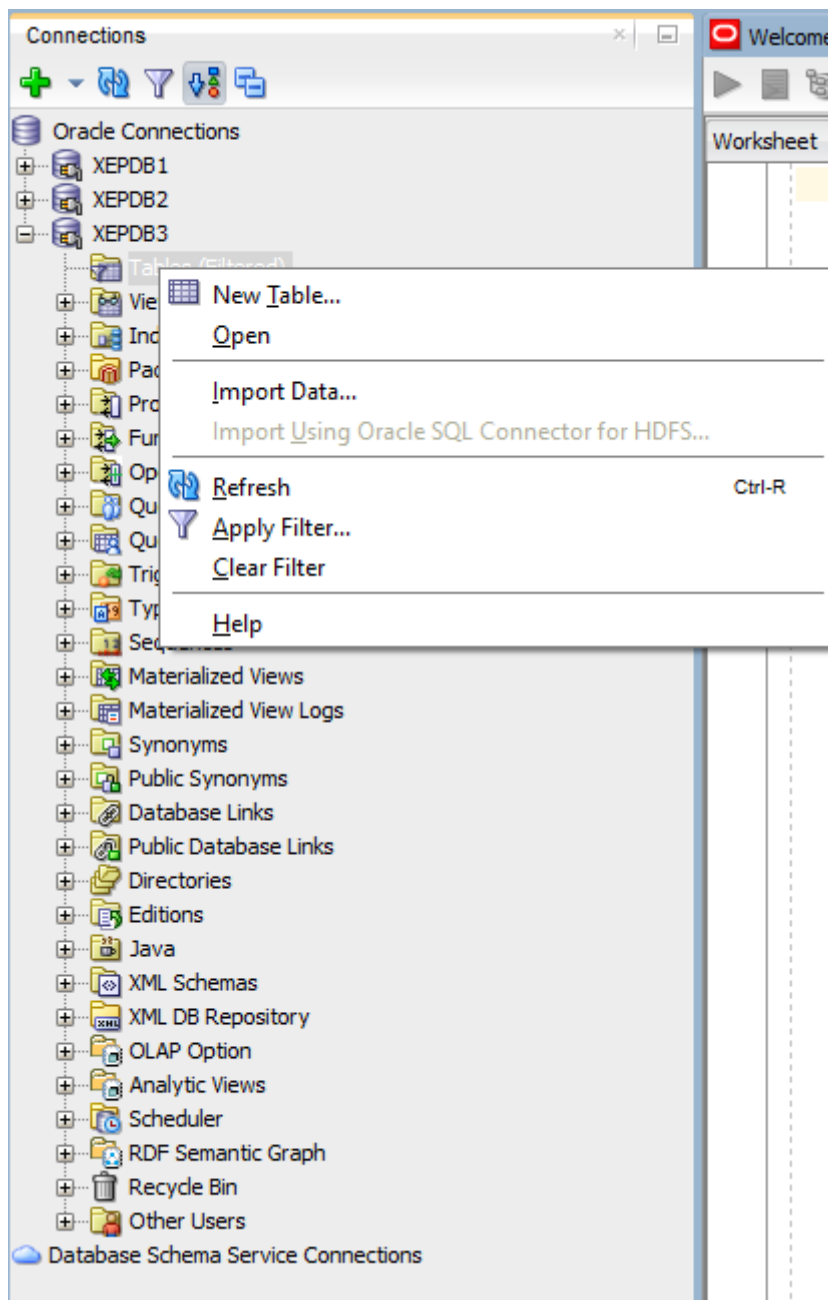
<https://eforexcel.com/wp/downloads-17-sample-csv-files-data-sets-for-testing-credit-card/>

W tym celu utworzymy nowe połączenie w SQL Developer:



The screenshot shows the 'New / Select Database Connection' dialog box in SQL Developer. The 'Name' field is set to 'XEPDB3'. The 'Database Type' is set to 'Oracle'. Under the 'User Info' tab, the 'Authentication Type' is 'Default', the 'Username' is 'pdb3_daniel', and the 'Role' is 'default'. The 'Password' field is masked with dots. The 'Save Password' checkbox is checked. Under the 'Connection Type' dropdown, 'Basic' is selected. Under the 'Details' tab, the 'Hostname' is 'localhost', the 'Port' is '3000', and the 'Service name' is 'XEPDB3'. The 'Status' at the bottom left is 'Success'. At the bottom, there are buttons for 'Pomoc', 'Save', 'Clear', 'Test', 'Connect', and 'Anuluj'.

Plik jest z rozszerzeniem csv, zatem jego wczytanie będzie się różnić od wczytania bazy danych w kontenerze XEPDB1. Prawym przyciskiem myszy klikamy na Tables i wybieramy Import Data...



Wybieramy lokalizację naszego pliku i zaznaczamy Header.

Data Preview

Data Preview

Import Method

Choose Columns

Column Definition

Finish

Source: Local File

File: C:\Users\jkal\Desktop\5m CC Records.csv

Browse...

Restore State

File Format

☒ Header

After Skip

Skip Rows: 0

Format: csv

☒ Preview Row Limit: 100

Encoding: Cp1250

Delimiter: ,

Line Terminator: standard: CR LF, CR or LF

Left Enclosure: "

Right Enclosure: "

File Contents

Card Type ...	Card Type F...	Issuing Bank	Card Number	Card Holder'...	CVV/CVV2	Issue Date	Expiry Date	Billing Date	Card PIN	Credit Limit
VI	Visa	U.S. Bancorp	4069039714...	Leon X Wilkie	482	04/2015	04/2023	27	2388	109600
AX	American Ex...	American Ex...	3443269397...	Denise R He...	8219	02/2010	02/2015	9	9250	12000
JC	Japan Credi...	JCB	3531335478...	Collin Curtis	227	11/2011	11/2030	10	8047	135100
DC	Diners Club ...	Diners Club	3001718424...	Aaron Estes	799	02/2016	02/2017	16	2879	120800
JC	Japan Credi...	JCB	3555003079...	Jacquelyn Kaya	794	09/2020	09/2038	3	6592	132100
JC	Japan Credi...	JCB	3565656781...	Tyler Melton	133	11/2020	11/2034	6	5207	133100
MC	Master Card	U.S. Bancorp	5203512736...	Jean Mcgarity	224	08/2016	08/2024	2	6978	13600
VI	Visa	Capital One	4149592666...	Catherine J ...	732	01/2018	01/2028	22	8188	187300
MC	Master Card	Bank of Ame...	5314960905...	Jannie Q Re...	759	12/2020	12/2029	16	6857	114300
DS	Discover	Discover	6011949126...	Derrick Boyer	210	08/2016	08/2025	17	8329	70000

Pomog

< Wstecz

Dalej >

Zakończ

Anuluj

Import Method

Data Preview

Import Method

Column Definition

Finish

Specify the method for importing data. For External Table method, an external table will be created to read the data in the file. For Staging External Table method, an external table will be created as a staging table for importing the target table. For other methods, a new table is created and the data is imported.

Import Method: Insert

☐ Send Create Script to SQL Worksheet

Table Name: CreditCard

☐ Import Row Limit: 100

File Contents

Card Type ...	Card Type F...	Issuing Bank	Card Number	Card Holder'...	CVV/CVV2	Issue Date	Expiry Date	Billing Date	Card PIN	Credit Limit
VI	Visa	U.S. Bancorp	4069039714...	Leon X Wilkie	482	04/2015	04/2023	27	2388	109600
AX	American Ex...	American Ex...	3443269397...	Denise R He...	8219	02/2010	02/2015	9	9250	12000
JC	Japan Credi...	JCB	3531335478...	Collin Curtis	227	11/2011	11/2030	10	8047	135100
DC	Diners Club ...	Diners Club	3001718424...	Aaron Estes	799	02/2016	02/2017	16	2879	120800
JC	Japan Credi...	JCB	3555003079...	Jacquelyn Kaya	794	09/2020	09/2038	3	6592	132100
JC	Japan Credi...	JCB	3565656781...	Tyler Melton	133	11/2020	11/2034	6	5207	133100
MC	Master Card	U.S. Bancorp	5203512736...	Jean Mcgarity	224	08/2016	08/2024	2	6978	13600
VI	Visa	Capital One	4149592666...	Catherine J ...	732	01/2018	01/2028	22	8188	187300
MC	Master Card	Bank of Ame...	5314960905...	Jannie Q Re...	759	12/2020	12/2029	16	6857	114300
DS	Discover	Discover	6011949126...	Derrick Boyer	210	08/2016	08/2025	17	8329	70000
AX	American Ex...	American Ex...	3716245144...	Taren W Solari	4718	12/2010	12/2030	19	8355	29700
AX	American Ex...	American Ex...	3468186469...	Esther Cross	7321	04/2016	04/2031	20	4323	135500
JC	Japan Credi...	JCB	3561700442...	Jeff F Maday	352	08/2019	08/2038	21	0155	136900
MC	Master Card	Capital One	5150744671...	Russell J Gr...	422	01/2011	01/2031	14	5830	67800
DS	Discover	Discover	6223751201...	Esteban R N...	882	11/2013	11/2031	6	6166	77900
VI	Visa	Chase	4482738091...	Melvin Town...	680	04/2012	04/2029	24	2445	104300
DS	Discover	Discover	6011979888...	Darell A Mabon	015	12/2017	12/2031	11	3622	128400

Pomog

< Wstecz

Dalej >

Zakończ

Anuluj

Choose Columns

Select the columns to import from the data set and arrange them in the order you want.

Available Columns

Selected Columns

Card Type Code
Card Type Full Name
Issuing Bank
Card Number
Card Holder's Name
CVV/CVV2
Issue Date
Expiry Date
Billing Date
Card PIN
Credit Limit

File Contents

Card Type ...	Card Type F...	Issuing Bank	Card Number	Card Holder'...	CVV/CVV2	Issue Date	Expiry Date	Billing Date	Card PIN	Credit Limit
VI	Visa	U.S. Bancorp	4069039714...	Leon X Wilkie	482	04/2015	04/2023	27	2388	109600
AX	American Ex...	American Ex...	3443269397...	Denise R He...	8219	02/2010	02/2015	9	9250	12000
JC	Japan Credi...	JCB	3531335478...	Collin Curtis	227	11/2011	11/2030	10	8047	135100

Pomoc < Wstecz Dalej > Zakończ Anuluj

Column Definition

For each column on left, define the column details of the database table that will be created to import this data into.

Source Data Columns

Card Type Code
Card Type Full Name
Issuing Bank
Card Number
Card Holder's Name
CVV/CVV2
Issue Date
Expiry Date
Billing Date
Card PIN
Credit Limit

Target Table Columns

Name CardTypeCode

Data Type VARCHAR2

Size/Precision 26

☒ Nullable? Default

Comment

Data

VI
AX
JC
DC
JC
JC
MC
VI
MC
DS
AX
AX

Status

Pomoc < Wstecz Dalej > Zakończ Anuluj

Finish

Finish

Save State

[Data Preview](#)
[Import Method](#)
[Choose Columns](#)
[Column Definition](#)
Finish

Import Summary
 Destination Connection: XEPDB3
 Source File: C:\Users\jkal\Desktop\5m CC Records.csv
 Selected Fields
 Fields Not Selected
 Import Method: Insert

Pomoc < Wstecz Dalej > Zakończ Anuluj

Przykładowe zapytanie dla sprawdzenia poprawności wczytania bazy danej Credit Card.

Connections

Oracle Connections

- XEPDB1
- XEPDB2
- XEPDB3

Tables (Filtered)

- CREDITCARD
 - CARDTYPECODE
 - CARDTYPEFULLNAME
 - ISSUINGBANK
 - CARDNUMBER
 - CARDHOLDERSNAME
 - CVVORCV2
 - ISSUEDATE
 - EXPIRYDATE
 - BILLINGDATE
 - CARDPIN
 - CREDITLIMIT

Views

- Indexes
- Packages
- Procedures
- Functions
- Operators
- Queues
- Queues Tables
- Triggers
- Types
- Sequences
- Materialized Views
- Materialized View Logs
- Synonyms
- Public Synonyms
- Database Links
- Public Database Links
- Directories
- Editions
- Java
- XML Schemas

Welcome Page XEPDB2 XEPDB3

Worksheet Query Builder

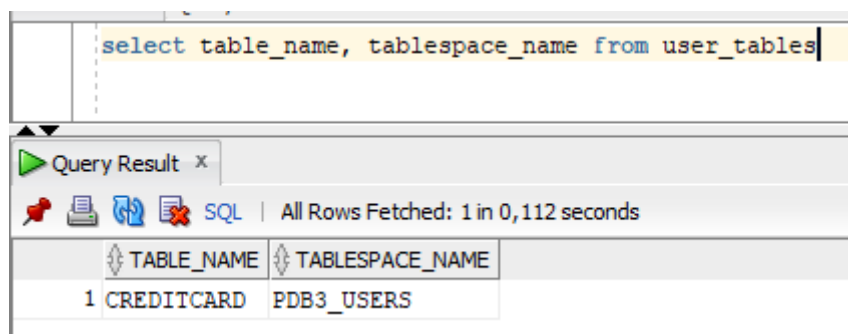
```
select cardtypefullname from creditcard
```

Query Result x

SQL | Fetched 50 rows in 0,006 seconds

CARDTYPEFULLNAME
1 Discover
2 Discover
3 Discover
4 Discover
5 Diners Club International
6 American Express
7 Visa
8 Japan Credit Bureau
9 Japan Credit Bureau
10 American Express
11 Diners Club International
12 American Express
13 Visa
14 Master Card
15 American Express
16 Diners Club International
17 Diners Club International
18 American Express
19 Japan Credit Bureau
20 Japan Credit Bureau
21 Japan Credit Bureau
22 Discover
23 American Express
24 American Express

Wyświetlamy przestrzeń tabeli CREDITCARD



4.5 Konfiguracja ustawień Oracle's National Language Support (NLS)

Za pomocą SQL Plus sprawdzimy obecne ustawienia NLS dla systemu bazodanowego, instancji bazy danych oraz sesji.

Zacznijmy od konteneru XEPDB1:

```
SQL> SELECT DB.PARAMETER, DB.VALUE "DATABASE", I.VALUE "INSTANCE", S.VALUE "SESSION" FROM NLS_DATABASE_PARAMETERS DB, NLS_INSTANCE_PARAMETERS I, NLS_SESSION_PARAMETERS S WHERE DB.PARAMETER=I.PARAMETER(+) AND DB.PARAMETER=S.PARAMETER(+) ORDER BY 1;
```

PARAMETER
DATABASE
INSTANCE
SESSION
NLS_CALENDAR
GREGORIAN

NLS_CURRENCY	NLS_DATE_FORMAT	NLS_DATE_LANGUAGE	NLS_LANGUAGE	NLS_TERRITORY
\$	DD-MON-RR	AMERICAN	AMERICAN	AMERICA
\$	DD-MON-RR	AMERICAN	POLISH	POLAND
\$	DD-MON-RR	AMERICAN	AMERICAN	AMERICA

Zmieniamy ustawienia NLS:

```
SQL> alter session set NLS_DATE_FORMAT = "day-month-YYYY";
Session altered.

SQL> select data_rej from samochody;
```

DATA_REJ
monday -march -2001
friday -september-2019
sunday -december -2006
sunday -april -2001
sunday -may -2007
sunday -july -2011
friday -february -2009
tuesday -january -2018
saturday -november -2017
tuesday -march -2019
friday -july -2009

DATA_REJ
tuesday -march -2013
friday -february -2020
thursday -may -1998
sunday -april -1999

Session modified.

ID_POJAZDU	ROCZNIK	CENA	VIN	DATA_REJ			
DATA_PRZ		POW	NR_REJ	SIL	MOC	ID_NADWOZIE	ID_PALIWA
ID_MODELU	ID_KOLORU	PRZEBIEG					ID_KRAJU
mardi	3	2005	12500	brak		dimanche	dÚcembre -2006
	-dÚcembre -2010	nie	RBR12323	2.0	90	3	1
							15

Session modifi e.

ID_POJAZDU	ROCZNIK		CENA		VIN	DATA_REJ	DATA_PRZ	POW
NR_REJ	SIL	MOC	ID_NADWOZIE	ID_PALIWA	ID_KRAJU	ID_MODELU	ID_KOLORU	
PRZEBIEG								
	2		1999	4500	brak	05/03/01	28/12/10	nie
KRA21312	1.9	60		2	3	1	2	3
184000								
	1		2019	210000	1FA545235KSDFS32	20/09/19	28/12/20	nie
KK 43232	2.0	210		1	1	10	3	4
199999								
ID_POJAZDU	ROCZNIK		CENA		VIN	DATA_REJ	DATA_PRZ	POW
NR_REJ	SIL	MOC	ID_NADWOZIE	ID_PALIWA	ID_KRAJU	ID_MODELU	ID_KOLORU	
PRZEBIEG								
	3		2005	12500	brak	17/12/06	28/12/10	nie
RBR12323	2.0	90		3	1	15	40	1
30250								

Obecny format wyświetlania stempla czasu.

```
SQL> SELECT CURRENT_TIMESTAMP FROM DUAL;

CURRENT_TIMESTAMP
-----
28/05/22 12:31:16,157000 +02:00
```

```
SQL> ALTER SESSION SET NLS_TIMESTAMP_FORMAT = 'HH:MI:SS.FF yyyy/mm/dd';

Session modified.
```

```
SQL> SELECT CAST(SYSTIMESTAMP AS TIMESTAMP) FROM DUAL;

CAST(SYSTIMESTAMP AS TIMESTAMP)
-----
12:33:18.924000 2022/05/28
```

Zmiana formatu kalendarza.

```
SQL> ALTER SESSION SET NLS_CALENDAR = 'Arabic Hijrah';

Session modified.

SQL> select to_char(sysdate, 'YYYY') as current_year from dual;

CURR
----
1443
```

Zmiana formatu waluty.

```
SQL> alter session set NLS_CURRENCY = "$";

Session modified.

SQL> SELECT TO_CHAR(cena, 'L99G999D99') cena FROM samochody;

CENA
-----
          $4 500,00
#####
          $12 500,00
          $4 200,00
          $12 555,00
          $21 500,00
          $34 100,00
#####
#####
#####
          $30 000,00
          $42 120,00
```

```
SQL> ALTER SESSION SET NLS_ISO_CURRENCY = "France";

Session modified.
```

Sprawdzimy raz jeszcze ustawienia NLS dla systemu bazodanowego, instancji bazy danych oraz sesji.

```
SQL> SELECT DB.PARAMETER, DB.VALUE "DATABASE", I.VALUE "INSTANCE", S.VALUE "SESSION" FROM NLS_DATABASE_PARAMETERS DB, NLS_INSTANCE_PARAMETERS I, NLS_SESSION_PARAMETERS S WHERE DB.PARAMETER=I.PARAMETER(+) AND DB.PARAMETER=S.PARAMETER(+) ORDER BY 1;
```

NLS_CALENDAR GREGORIAN	NLS_CURRENCY \$	NLS_DATE_FORMAT DD-MON-RR	NLS_DATE_LANGUAGE AMERICAN
Arabic Hijrah	\$	DD Month YYYY	FRENCH
NLS_LANGUAGE AMERICAN POLISH FRENCH	NLS_TERRITORY AMERICA POLAND FRANCE	NLS_TIMESTAMP_FORMAT DD-MON-RR HH.MI.SSXXFF AM HH:MI:SS.FF yyyy/mm/dd	

Przejdźmy teraz do konteneru XEPDB2.

```
SQL> SELECT DB.PARAMETER, DB.VALUE "DATABASE", I.VALUE "INSTANCE", S.VALUE "SESSION" FROM NLS_DATABASE_PARAMETERS DB, NLS_INSTANCE_PARAMETERS I, NLS_SESSION_PARAMETERS S WHERE DB.PARAMETER=I.PARAMETER(+) AND DB.PARAMETER=S.PARAMETER(+) ORDER BY 1;
```

```

PARAMETER
-----
DATABASE
-----
INSTANCE
-----
SESSION
-----
NLS_CALENDAR
GREGORIAN
-----
GREGORIAN
-----

```

NLS_CURRENCY	NLS_DATE_FORMAT	NLS_DATE_LANGUAGE	NLS_LANGUAGE	NLS_TERRITORY
\$	DD-MON-RR	AMERICAN	AMERICAN	AMERICA
			POLISH	POLAND
\$	DD-MON-RR	AMERICAN	AMERICAN	AMERICA

Zmieńmy ustawienia NLS

```
SQL> ALTER SESSION SET NLS_DATE_FORMAT = "rr-mm-dd";
```

Session altered.

```
SQL> select END_OF_CAL_YEAR FROM TIMES;
```

END OF C

[illegible]


```
SQL> ALTER SESSION SET NLS_LANGUAGE = POLISH;

Zmieniono sesje.

SQL> SELECT * FROM CUSTOMERS;

  CUST_ID CUST_FIRST_NAME      CUST_LAST_NAME          C
-----
CUST_YEAR_OF_BIRTH CUST_MARITAL_STATUS  CUST_STREET_ADDRESS
-----
CUST_POSTA CUST_CITY              CUST_CITY_ID
-----
CUST_STATE_PROVINCE              CUST_STATE_PROVINCE_ID COUNTRY_ID
-----
CUST_MAIN_PHONE_NUMBER  CUST_INCOME_LEVEL      CUST_CREDIT_LIMIT
-----
CUST_EMAIL              CUST_TOTAL      CUST_TOTAL_ID
-----
CUST_SRC_ID CUST_EFF CUST_EFF C
-----
49671 Abigail          Ruddy          M
1976 married          27 North Sagadahoc Boulevard
```

```
SQL> ALTER SESSION SET NLS_TERRITORY = POLAND;

Zmieniono sesje.

SQL> SELECT * FROM CUSTOMERS;

  CUST_ID CUST_FIRST_NAME      CUST_LAST_NAME          C
-----
CUST_YEAR_OF_BIRTH CUST_MARITAL_STATUS  CUST_STREET_ADDRESS
-----
CUST_POSTA CUST_CITY              CUST_CITY_ID
-----
CUST_STATE_PROVINCE              CUST_STATE_PROVINCE_ID COUNTRY_ID
-----
CUST_MAIN_PHONE_NUMBER  CUST_INCOME_LEVEL      CUST_CREDIT_LIMIT
-----
CUST_EMAIL              CUST_TOTAL      CUST_TOTAL_ID
-----
CUST_SRC_ID CUST_EFF CUST_EFF C
-----
49671 Abigail          Ruddy          M
1976 married          27 North Sagadahoc Boulevard
60332 Ede              51442
Gelderland              52610      52770
519-236-6123      G: 130,000 - 149,999      1500
Ruddy@company.example.com      Customer total      52772
98/01/01          I
```

Obecny format wyświetlania stempla czasu.

```
SQL> SELECT CURRENT_TIMESTAMP FROM DUAL;

CURRENT_TIMESTAMP
-----
28/05/22 12:31:16,157000 +02:00
```

```
SQL> ALTER SESSION SET NLS_TIMESTAMP_FORMAT = 'HH:MI:SS.FF rr-mm-yy';
Zmieniono sesje.
SQL> SELECT CAST(SYSTIMESTAMP AS TIMESTAMP) FROM DUAL;
CAST(SYSTIMESTAMP AS TIMESTAMP)
-----
12:59:59.757000 22-05-22
Wybrano wierszy: 1.
```

Zmiana formatu kalendarza

```
SQL> ALTER SESSION SET NLS_CALENDAR = 'Gregorian';
Zmieniono sesje.
SQL> SELECT TO_CHAR(SYSDATE, 'YYYY') AS CURRENT_YEAR FROM DUAL;
CURRENT_YEAR
-----
2022
```

Zmiana formatu waluty

```
SQL> ALTER SESSION SET NLS_CURRENCY = "zł";
Zmieniono sesje.
SQL> SELECT TO_CHAR(UNIT_PRICE, 'L99G999D99') cena FROM COSTS;
CENA
-----
      zł 176,23
      zł 013,99
      zł999,99
      zł999,99
      zł 003,99
      zł 003,49
      zł 632,79
      zł 655,65
      zł 697,83
      zł62,37
      zł62,15
      zł62,15
      zł63,02
```

```
SQL> ALTER SESSION SET NLS_ISO_CURRENCY = "Poland";
Zmieniono sesje.
```

Sprawdzimy raz jeszcze ustawienia NLS dla systemu bazodanowego, instancji bazy danych oraz sesji.

```
SQL> SELECT DB.PARAMETER, DB.VALUE "DATABASE", I.VALUE "INSTANCE", S.VALUE "SESSION" FROM NLS_DATABASE_PARAMETERS DB, NLS_INSTANCE_PARAMETERS I, NLS_SESSION_PARAMETERS S WHERE DB.PARAMETER=I.PARAMETER(+) AND DB.PARAMETER=S.PARAMETER(+) ORDER BY 1;
```

NLS_CALENDAR GREGORIAN	NLS_CURRENCY \$	NLS_DATE_FORMAT DD-MON-RR	NLS_DATE_LANGUAGE AMERICAN
GREGORIAN	zł	RR/MM/DD	POLISH
NLS_LANGUAGE AMERICAN	NLS_TERRITORY AMERICA	NLS_TIMESTAMP_FORMAT DD-MON-RR HH.MI.SSXF AM	
POLISH	POLAND	HH:MI:SS.FF rr-mm-yy	
POLISH	POLAND		

Przejdźmy teraz do konteneru XEPDB3.

```
SQL> SELECT DB.PARAMETER, DB.VALUE "DATABASE", I.VALUE "INSTANCE", S.VALUE "SESSION" FROM NLS_DATABASE_PARAMETERS DB, NLS_INSTANCE_PARAMETERS I, NLS_SESSION_PARAMETERS S WHERE DB.PARAMETER=I.PARAMETER(+) AND DB.PARAMETER=S.PARAMETER(+) ORDER BY 1;

PARAMETER
-----
DATABASE
-----
INSTANCE
-----
SESSION
-----
NLS_CALENDAR
GREGORIAN
GREGORIAN
```

NLS_CURRENCY	NLS_DATE_FORMAT	NLS_DATE_LANGUAGE	NLS_LANGUAGE	NLS_TERRITORY
\$	DD-MON-RR	AMERICAN	AMERICAN	AMERICA
\$	DD-MON-RR	AMERICAN	POLISH	POLAND
			AMERICAN	AMERICA

Zmieńmy ustawienia NLS:

```
SQL> ALTER SESSION SET NLS_DATE_FORMAT = "dd.mm.yyyy";

Session altered.

SQL> SELECT ISSUEDATE FROM CREDITCARD;

ISSUEDATE
-----
20.09.2019
20.09.2015
20.02.2010
20.06.2015
20.02.2019
20.03.2011
20.10.2016
```

```
SQL> ALTER SESSION SET NLS_LANGUAGE = AMERICAN;

Session altered.

SQL> SELECT * from CREDITCARD;

CARDTYPECODE      CARDTYPEFULLNAME      ISSUINGBANK      CARDNUMBER  CARDHOLDERSNAME
CVVORCVV2          ISSUEDATE  EXPIRYDATE  BILLINGDATE
-----
CARDPIN            CREDITLIMIT
-----
DS                 Discover
234               20.09.2019  20.09.2034  Discover      6.0116E+15  Danielle N Sheppard
3870              48700
DS                 Discover
730               20.09.2015  20.09.2033  Discover      6.0117E+15  Genaro R Wenger
5321              150400
DS                 Discover
860               20.02.2010  20.02.2011  Discover      6.2255E+15  Ruby U Shannon
7659              78600
```

```
SQL> ALTER SESSION SET NLS_TERRITORY = AMERICA;
Session altered.
SQL> SELECT * from CREDITCARD;
```

CARDTYPECODE CVVORCVV2	CARDTYPEFULLNAME ISSUEDATE EXPIRYDAT	ISSUINGBANK BILLINGDATE	CARDNUMBER	CARDHOLDERSNAME
DS 234 3870	Discover 20-SEP-19 20-SEP-34 48700	Discover 26	6.0116E+15	Danielle N Sheppard
DS 730 5321	Discover 20-SEP-15 20-SEP-33 150400	Discover 6	6.0117E+15	Genaro R Wenger
DS 860	Discover 20-FEB-10 20-FEB-11	Discover 2	6.2255E+15	Ruby U Shannon

Obecny format wyświetlania stempla czasu.

```
SQL> SELECT CURRENT_TIMESTAMP FROM DUAL;
CURRENT_TIMESTAMP
-----
28/05/22 12:31:16,157000 +02:00
```

```
SQL> ALTER SESSION SET NLS_TIMESTAMP_FORMAT = 'HH:MI:SS.FF dd.mm.yyyy';
Session altered.
SQL> SELECT CAST(SYSTIMESTAMP AS TIMESTAMP) FROM DUAL;
CAST(SYSTIMESTAMP AS TIMESTAMP)
-----
01:19:44.271000 28.05.2022
```

Zmiana formatu kalendarza

```
SQL> ALTER SESSION SET NLS_CALENDAR = 'Thai Buddha';
Session altered.
SQL> SELECT TO_CHAR(SYSDATE, 'YYYY') AS CURRENT_YEAR FROM DUAL;
CURRENT_YEAR
-----
2565
```

Zmiana formatu waluty

```
SQL> ALTER SESSION SET NLS_CURRENCY = "€";
Session altered.
SQL> SELECT TO_CHAR(CREDITLIMIT, 'L99G999D99') cena from CREDITCARD;
CENA
-----
L48,700.00
#####
L78,600.00
L57,500.00
L10,200.00
L92,100.00
L62,300.00
```

```
SQL> ALTER SESSION SET NLS_ISO_CURRENCY = "AMERICA";
Session altered.
```

Sprawdzimy raz jeszcze ustawienia NLS dla systemu bazodanowego, instancji bazy danych oraz sesji.

```
SQL> SELECT DB.PARAMETER, DB.VALUE "DATABASE", I.VALUE "INSTANCE", S.VALUE "SESSION" FROM NLS_DATABASE_PARAMETERS DB, NLS_INSTANCE_PARAMETERS I, NLS_SESSION_PARAMETERS S WHERE DB.PARAMETER=I.PARAMETER(+) AND DB.PARAMETER=S.PARAMETER(+) ORDER BY 1;
```

NLS_CALENDAR GREGORIAN	NLS_CURRENCY \$	NLS_DATE_FORMAT DD-MON-RR	NLS_DATE_LANGUAGE AMERICAN
Thai Buddha	L	"דגדג" dd month EE yyyy	AMERICAN

NLS_ISO_CURRENCY AMERICA	NLS_LANGUAGE AMERICAN	NLS_TERRITORY AMERICA
AMERICA	POLISH	POLAND
	AMERICAN	AMERICA

4.6 Unplug and Plug a PDB

4.6.1 Na tej samej maszynie

Wyświetlamy wszystkie PDB

```
SQL> show pdbs;
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	NO
3	XEPDB1	READ WRITE	NO
4	XEPDB2	READ WRITE	NO
5	XEPDB3	READ WRITE	NO

```
SQL> _
```

W tym przypadku odłączymy bazę danych XEPDB3, a następnie podłączymy ją pod nazwą XEPDB5.

Połączymy się bazą danych XEPDB3 i wyświetlimy dba_data_files.

```
SQL> alter session set container = XEPDB3;
Session altered.

SQL> select file_name from dba_data_files;
```

FILE_NAME
C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB3\UNDOTBS01.DBF
C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB3\SYS_AUX01.DBF
C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB3\SYSTEM01.DBF
C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB3\PDB3_USERS.DBF

```
4 rows selected.
```

Zamykamy XEPDB3.

```
SQL> connect system/system as sysdba
Connected.
SQL> alter pluggable database XEPDB3 close immediate;

Pluggable database altered.

SQL> show pdbs;

  CON_ID CON_NAME                                OPEN MODE  RESTRICTED
-----
      2 PDB$SEED                                READ ONLY  NO
      3 XEPDB1                                  READ WRITE NO
      4 XEPDB2                                  READ WRITE NO
      5 XEPDB3                                  MOUNTED
```

Odłącz zamknięty plik PDB, a następnie podaj ścieżkę i nazwę pliku XML. Usuń zamknięty plik PDB i zachowaj pliki danych.

```
SQL> ALTER PLUGGABLE DATABASE XEPDB3 UNPLUG INTO 'C:/app/dkcal/product/18.0.0/oradata/XE/XEPDB3/XEPDB3.xml';

Pluggable database altered.

SQL> DROP PLUGGABLE DATABASE XEPDB3 KEEP DATAFILES;

Pluggable database dropped.

SQL> show pdbs;

  CON_ID CON_NAME                                OPEN MODE  RESTRICTED
-----
      2 PDB$SEED                                READ ONLY  NO
      3 XEPDB1                                  READ WRITE NO
      4 XEPDB2                                  READ WRITE NO
```

```
SQL> CREATE PLUGGABLE DATABASE XEPDB5 USING 'C:\app\dkcal\product\18.0.0\oradata\XE\XEPDB3\XEPDB3.xml' FILE_NAME_CONVERT=('C:\app\dkcal\product\18.0.0\oradata\XE\XEPDB3','C:\app\dkcal\product\18.0.0\oradata\XE\XEPDB5');

Pluggable database created.

SQL> show pdbs;

  CON_ID CON_NAME                                OPEN MODE  RESTRICTED
-----
      2 PDB$SEED                                READ ONLY  NO
      3 XEPDB1                                  READ WRITE NO
      4 XEPDB2                                  READ WRITE NO
      5 XEPDB5                                  MOUNTED
```

Otwieramy XEPDB5.

```
SQL> alter pluggable database XEPDB5 open;
Pluggable database altered.

SQL> show pdbs;

  CON_ID CON_NAME                                OPEN MODE RESTRICTED
-----
      2 PDB$SEED                                READ ONLY NO
      3 XEPDB1                                READ WRITE NO
      4 XEPDB2                                READ WRITE NO
      5 XEPDB5                                READ WRITE NO

SQL>
```

W pliku tsnames.ora zmieniamy XEPDB3 na XEPDB5.

```
XEPDB5 =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 3000))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = XEPDB5)
    )
  )
```

```
SQL> connect pdb3_daniel/daniel@XEPDB5
Connected.
SQL> show con_name;

CON_NAME
-----
XEPDB5
SQL> show user;
USER is "PDB3_DANIEL"
SQL>
```

```
SQL> select * from CREDITCARD;
```

CARDTYPECODE	CARDTYPEFULLNAME	ISSUINGBANK
CARDNUMBER	CARDHOLDERSNAME	CVVORCVV2
EXPIRYDAT	BILLINGDATE	CARDPIN
		CREDITLIMIT
DS	Discover	Discover
6.0116E+15	Danielle N Sheppard	234
20-SEP-34	26 3870	48700
DS	Discover	Discover
6.0117E+15	Genaro R Wenger	730
20-SEP-33	6 5321	150400
		20-SEP-19

```
SQL> select name from v$datafile where con_id = 5;

NAME
-----
C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB5\SYSTEM01.DBF
C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB5\SYSAUX01.DBF
C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB5\UNDOTBS01.DBF
C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB5\PDB3_USERS.DBF
```

```
SQL> connect pdb3_daniel/daniel@localhost:3000/XEPDB5
Connected.
SQL> show con_name;

CON_NAME
-----
XEPDB5
SQL> show user;
USER is "PDB3_DANIEL"
SQL> connect pdb3_pawel/pawel@localhost:3000/XEPDB5
Connected.
SQL>
```

Jak widać wszystko zostało wczytane poprawnie, wszystko to co było w kontenerze XEPDB3 znajduje się w XEPDB5.

4.6.2 Na drugiej maszynie wirtualnej

Wyświetlamy wszystkie PDB

```
SQL> show pdbs

  CON_ID CON_NAME          OPEN MODE  RESTRICTED
  -----
      2 PDB$SEED              READ ONLY  NO
      3 XEPDB1              MOUNTED
      4 XEPDB2              MOUNTED
```

Utworzenie kontenera PDB na podstawie pliku xml

```
SQL> CREATE PLUGGABLE DATABASE PDB3 USING 'C:\app\IEUser\product\18.0.0\oradata\XE\XEPDB3\XEPDB3.xml' SOURCE_FILE_NAME_CONVERT=('C:\APP\DKCAL\PRODUCT\18.0.0\ORADATA\XE\XEPDB3','C:\app\IEUser\product\18.0.0\oradata\XE\XEPDB3') nocopy tempfile reuse;

Pluggable database created.

SQL> show pdbs;

  CON_ID CON_NAME          OPEN MODE  RESTRICTED
  -----
      2 PDB$SEED              READ ONLY  NO
      3 XEPDB1              MOUNTED
      4 XEPDB2              MOUNTED
      5 PDB3                MOUNTED
```

Otwieramy PDB3.


```
SQL> alter pluggable database PDB3 open;
```

```
Pluggable database altered.
```

```
SQL> show pdbs;
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	NO
3	XEPDB1	MOUNTED	
4	XEPDB2	MOUNTED	
5	PDB3	READ WRITE	NO

```
SQL>
```

Dodanie aliasu PDB3

tnsnames.ora - Notepad

File Edit Format View Help

```
LISTENER_XEPDB3 =  
  (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 3000))  
  
XE =  
  (DESCRIPTION =  
    (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))  
    (CONNECT_DATA =  
      (SERVER = DEDICATED)  
      (SERVICE_NAME = XE)  
    )  
  )  
  
XEPDB1 =  
  (DESCRIPTION =  
    (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1000))  
    (CONNECT_DATA =  
      (SERVER = DEDICATED)  
      (SERVICE_NAME = XEPDB1)  
    )  
  )  
  
XEPDB2 =  
  (DESCRIPTION =  
    (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 2000))  
    (CONNECT_DATA =  
      (SERVER = DEDICATED)  
      (SERVICE_NAME = XEPDB2)  
    )  
  )  
  
PDB3 =  
  (DESCRIPTION =  
    (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 3000))  
    (CONNECT_DATA =  
      (SERVER = DEDICATED)  
      (SERVICE_NAME = PDB3)|  
    )  
  )  
)
```

Przełączenie się na kontener PDB3 i wyświetlenie użytkowników.

```
SQL> alter session set CONTAINER = PDB3;
Session altered.
SQL> show con_name;
CON_NAME
-----
PDB3
SQL> select username from dba_users where username like 'PDB3_%';
USERNAME
-----
PDB3_PAWEŁ
PDB3_ADMIN
PDB3_DANIEL
```

Przełączanie się między użytkownikami.

```
SQL> connect pdb3_daniel/daniel@PDB3
Connected.
SQL> show con_name;
CON_NAME
-----
PDB3
SQL> show user;
USER is "PDB3_DANIEL"
SQL> connect pdb3_pawel/pawel@PDB3;
Connected.
SQL> show con_name;
CON_NAME
-----
PDB3
SQL> show user;
USER is "PDB3_PAWEŁ"
SQL> connect pdb3_daniel/daniel@localhost:3000/PDB3
Connected.
SQL> show user;
USER is "PDB3_DANIEL"
SQL> connect pdb3_pawel/pawel@localhost:3000/PDB3
Connected.
SQL> show user;
USER is "PDB3_PAWEŁ"
SQL>
```

Sprawdzamy plik file_name.

```

SQL> connect pdb3_admin/oracle@localhost:3000/PDB3
Connected.
SQL> show con_name;

CON_NAME
-----
PDB3
SQL> show user;
USER is "PDB3_ADMIN"
SQL> select file_name from dba_data_files;

FILE_NAME
-----
C:\APP\IEUSER\PRODUCT\18.0.0\ORADATA\XE\XEPDB3\PDB3_USERS.DBF
C:\APP\IEUSER\PRODUCT\18.0.0\ORADATA\XE\XEPDB3\UNDOTBS01.DBF
C:\APP\IEUSER\PRODUCT\18.0.0\ORADATA\XE\XEPDB3\SYSAUX01.DBF
C:\APP\IEUSER\PRODUCT\18.0.0\ORADATA\XE\XEPDB3\SYSTEM01.DBF

```

Sprawdzamy czy baza się wczytała.

```

SQL> select * from CREDITCARD;

CARDTYPECODE          CARDTYPEFULLNAME          ISSUINGBANK
-----
CARDNUMBER  CARDHOLDERSNAME          CVVORCVV2          ISSUEDATE
-----
EXPIRYDAT  BILLINGDATE  CARDPIN          CREDITLIMIT
-----
DS          Discover          Discover
6.0116E+15  Danielle N Sheppard    234                20-SEP-19
20-SEP-34   26 3870           48700
DS          Discover          Discover
6.0117E+15  Genaro R Wenger        730                20-SEP-15
20-SEP-33   6 5321           150400

```

Wyświetlenie przestrzeni tabel dla wczytanej bazy.

```

SQL> select table_name, tablespace_name from user_tables;

TABLE_NAME
-----
TABLESPACE_NAME
-----
CREDITCARD
PDB3_USERS

```