# Ćwiczenie 7. Retrospekcja

# 1. Sprawdzenie środowiska

Czas trwania: 15 minut

Mechanizmy związane z retrospekcją wymagają spełnienia kilku warunków. W pierwszym ćwiczeniu dokonamy ich weryfikacji.

Zakłada się przy tym, że:

- maszyna wirtualna wykorzystywana podczas ćwiczeń z administracji jest włączona w przeciwnym wypadku uruchom ją,
- baza danych zainstalowana na maszynie wirtualnej jest włączona w przeciwnym wypadku uruchom ją.
- proces nasłuchu jest włączony w przeciwnym wypadku uruchom go.
- 1. Uruchom terminal i zaloguj się do bazy danych jako użytkownik administracyjny.
- 2. Na początku sprawdź, czy baza danych działa w trybie archiwizacji plików dziennika powtórzeń oraz czy jest włączona opcja retrospekcji bazy danych.

SELECT log\_mode, flashback\_on FROM v\$database;

SQL> select LOG_MODE, FLASHBACK_ON from v\$database;	≡
LOG_MODE	
FLASHBACK_ON	
NOARCHIVELOG NO	
SQL>	~

- Powyższy wynik świadczy o tym, że baza danych działa bez archiwizacji plików dziennika powtórzeń (LOG\_MODE=NOARCHIVELOG) oraz, że opcja retrospekcji bazy danych jest wyłączona (FLASHBACK\_ON=NO).
- 4. Jeśli ćwiczenia z archiwizacji i odtwarzania zostały wykonane przez Ciebie poprawnie, Twoja baza danych powinna działać w trybie archiwizacji dziennika powtórzeń (LOG\_MODE=ARCHIVELOG). Jeśli tak nie jest, skonfiguruj opcję archiwizacji plików dziennika powtórzeń, w przeciwnym przypadku przejdź do następnego punktu.
  - a. Na początku zamknij bazę danych

SHUTDOWN IMMEDIATE;

b. Uruchom bazę danych w trybie mount.

STARTUP MOUNT;

c. Określ miejsce składowania plików dziennika powtórzeń.

```
ALTER SYSTEM SET
log_archive_dest_1='location=USE_DB_RECOVERY_FILE_DEST'
SCOPE=SPFILE;
```

d. Określ format nazwy plików dziennika powtórzeń.

```
ALTER SYSTEM SET
log_archive_format='ARC%S_%R.%T'
SCOPE=SPFILE;
```

e. Zamknij bazę danych i uruchom ją ponownie w trybie MOUNT.

show parameter DB\_RECOVERY\_FILE\_DEST;

SHUTDOWN IMMEDIATE;

STARTUP MOUNT;

f. Przełącz bazę danych w tryb archiwizacji plików dziennika powtórzeń.

ALTER DATABASE ARCHIVELOG;

g. Otwórz bazę danych.

ALTER DATABASE OPEN;

- 5. Skonfiguruj opcję retrospekcji bazy danych.
  - a. Na początku sprawdź gdzie znajduje się i jaki jest rozmiar domyślnej lokalizacji obszaru *flash recovery area*. Zawiera on z reguły kopie bieżących plików kontrolnych, bieżące i zarchiwizowane pliki dziennika powtórzeń, a także logi retrospekcji i archiwa RMAN-a.

SQL> show parameter DB_RECOVERY_FILE_DEST;	
NAME TYPE	
VALUE	
db_recovery_file_dest string /u01/app/oracle/flash_recovery	
_area	=
db_recovery_file_dest_size big int	eger
3852M	
SQL>	~

b. Następnie sprawdź jaki jest górny limit (w minutach) wycofania w czasie bazy danych (*flashback database*).

show parameter DB\_FLASHBACK\_RETENTION\_TARGET;

c. Wyłącz bazę danych.

SHUTDOWN IMMEDIATE;

d. Uruchom bazę danych w trybie MOUNT EXCLUSIVE. Włącz opcję retrospekcji bazy danych. Otwórz bazę danych

STARTUP MOUNT EXCLUSIVE;

ALTER DATABASE FLASHBACK ON;

ALTER DATABASE OPEN;

- 6. Na zakończenie sprawdzimy jeszcze parametry dotyczące przestrzeni tabel UNDO. Dane UNDO będą nam potrzebne miedzy innymi podczas korzystania z perspektywy flashback\_transaction\_query.
  - a. Sprawdź wartość parametru UNDO\_RETENTION.

show parameter UNDO\_RETENTION;

NAME	ТҮРЕ	
VALUE		
undo_retention 900	integer	=

- Wartość parametru UNDO\_RETENTION wyrażona jest w sekundach. Ile minut w Twoim przypadku system będzie utrzymywał dane UNDO po zakończeniu transakcji?
- b. Sprawdzimy teraz, czy utrzymywanie tych danych przez powyższy czas jest gwarantowane. Na początku jednak sprawdź jak nazywa się przestrzeń tabel UNDO wykorzystywana w Twojej bazie danych.

```
show parameter UNDO_TABLESPACE;
```

```
SELECT tablespace_name, retention
FROM dba_tablespaces
WHERE tablespace_name = 'UNDOTBS1';
```

		0	racle@loc	alhost:~		
<u>P</u> lik	<u>E</u> dycja	<u>W</u> idok	Terminal	Zakła <u>d</u> ki	Pomo <u>c</u>	
SQL>	show p	aramete	er UNDO_T	ABLESPACE	;	
NAME					TYPE	
VALU	 E					
undo	tables	nace			string	
UNDO	TBS1	puee			a crang	
SQL>	SELECT	tables	pace_nam	e, retent	ion	
2	FROM	dba_ta	blespace	S		
3	WHERE	tables	pace_nam	e = 'UNDO	TBS1';	
TABL	ESPACE_I	NAME				
RETE	NTION					
UNDO	TBS1					
NOGU	ARANTEE					

c. Jeśli wartość kolumny RETENTION w perspektywie dba\_tablespaces dla przestrzeni tabel UNDO posiada wartość NOGUARANTEE, dokonamy stosownej modyfikacji.

ALTER TABLESPACE UNDOTBS1 RETENTION GUARANTEE;

d. Na zakończenie przygotowań sprawdzimy czy na poziomie bazy danych włączone jest rejestrowanie dodatkowych informacji wymaganych między innymi dla retrospekcji transakcji.

```
SELECT supplemental_log_data_min
FROM v$database;

SQL> SELECT supplemental_log_data_min
2 FROM v$database;
SUPPLEMENTAL_LOG_DATA_MI
.....
N0
SQL>
```

e. Jeśli poniższe zapytanie zwróciło negatywny, włączymy rejestrowanie dodatkowych danych za pomocą poniższych poleceń.

ALTER DATABASE ADD SUPPLEMENTAL LOG DATA;

ALTER DATABASE ADD SUPPLEMENTAL LOG DATA (PRIMARY KEY) COLUMNS;

### 2. Flashback Query

Czas trwania: 10 minut

Pierwszym z testowanych mechanizmów będą zapytania retrospekcyjne.

1. Uruchom nowy terminal. Zaloguj się do bazy danych jako użytkownik SCOTT. Aby sesja użytkownika SCOTT była łatwiej rozpoznawalna, ustaw znak zachęty.

set sqlprompt SCOTT>

```
[oracle@localhost ~]$ sqlplus scott/tiger
SQL*Plus: Release 11.2.0.1.0 Production on Cz Mar 24 15:06:55 2011
Copyright (c) 1982, 2009, Oracle. All rights reserved.
Połączono z:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
SQL> set sqlprompt SCOTT>
SCOTT>
```

2. Jako użytkownik SCOTT utwórz tabelę test1.

CREATE TABLE test1 (id NUMBER);

3. W sesji administratora bazy danych, dla łatwiejszego jej rozpoznawania ustaw znak zachęty.

```
set sqlprompt ADMINISTRATOR>
```

4. Jako administrator bazy danych wykonaj zapytanie sprawdzające bieżący numer SCN oraz bieżący czas.

- Zapisz wartość numeru SCN
- Zapisz odczytany czas
- 5. Odczekaj dwie minuty, a następnie jako SCOTT wykonaj następujące operacje.

```
INSERT INTO test1 (id) VALUES (1);
```

COMMIT;

```
SELECT COUNT(*) FROM test1;
```

Zwróć uwagę na fakt iż transakcja, która modyfikowała zawartość danych została zakończona.

- Zapisz liczbę odczytanych wierszy
- 6. Wykorzystując mechanizm zapytań retrospekcyjnych przeglądnij zawartość tabeli test1 z przeszłości. Wykorzystaj w tym celu odczytaną wcześniej datę.

```
SELECT COUNT(*) FROM test1 AS OF TIMESTAMP
TO_TIMESTAMP('2011-03-24 15:25:26', 'YYYY-MM-DD HH24:MI:SS');
```

• Czy liczba wierszy odpowiadała zawartości tabeli test ze wskazanego momentu?

7. Sprawdź czy taki sam efekt uzyskany zostanie, jeżeli w zapytaniu użyty będzie numer SCN.

```
SELECT COUNT(*) FROM test1 AS OF SCN 834824;
SCOTT>SELECT COUNT(*) FROM test1 AS OF SCN 834824;
COUNT(*)
0
```

3. Flashback Version Query

Czas trwania: 5 minut

Mechanizm retrospekcji pozwala także na przegląd wersji danych jakie miały miejsce w przedziale czasu.

Jeśli chcesz, możesz zmienić narzędzie pracy dla użytkownika SCOTT z *SQL\*Plusa* na *SQL Developer*. Wykorzystanie narzędzia *SQL Developer* pozwoli w sposób prostszy przeglądać wyniki zapytań. Jeśli zdecydujesz się na wykorzystanie narzędzia *SQL Developer* zaloguj się za jego pomocą jako użytkownik SCOTT.

1. Jako użytkownik SCOTT wykonaj następujące polecenia.

CREATE TABLE test2 ( id NUMBER, opis VARCHAR(50)); INSERT INTO test2 (id, opis) VALUES (1, 'wartość pierwsza'); COMMIT;

2. Jako użytkownik administracyjny odczytaj bieżące wartości numeru SCN i czasu.

```
SELECT current_scn,
TO_CHAR(SYSTIMESTAMP, 'YYYY-MM-DD HH24:MI:SS')
FROM v$database;
```

- Zanotuj wartość numeru SCN
- Zanotuj odczytany czas

Powyższe wartości będą momentem, OD którego będziemy analizowali zmiany w tabeli test2.

3. Jako użytkownik SCOTT wykonaj kolejne polecenia.

```
UPDATE test2 SET opis = 'wartość druga' WHERE id = 1;
COMMIT;
```

UPDATE test2 SET opis = 'wartość trzecia' WHERE id = 1;

COMMIT;



4. Ponownie jako użytkownik administracyjny odczytaj bieżące wartości numeru SCN i czasu.

```
SELECT current_scn,
TO_CHAR(SYSTIMESTAMP, 'YYYY-MM-DD HH24:MI:SS')
FROM v$database;
```

- Zanotuj wartość numeru SCN
- Zanotuj odczytany czas

Tym razem wartości te będą dotyczyły momentu, DO którego będziemy analizowali zmiany w tabeli test2.

5. Czas na wykorzystanie mechanizmu retrospekcji. Jako użytkownik SCOTT zadaj zapytanie analizujące postać rekordu z tabeli test2 w wyznaczonym przedziale czasu. Wykorzystaj w zapytaniu czas zapisany przez Ciebie we wcześniejszych zadaniach.

```
SELECT versions_startscn, versions_starttime,
versions_endscn, versions_endtime,
versions_xid, versions_operation, opis
FROM test2
VERSIONS BETWEEN TIMESTAMP
TO_TIMESTAMP('2011-03-24 16:34:37', 'YYYY-MM-DD HH24:MI:SS')
AND
TO_TIMESTAMP('2011-03-24 17:04:17', 'YYYY-MM-DD HH24:MI:SS')
WHERE id = 1;
```

Jeśli pracujesz w SQL\*Plus, przed wykonaniem powyższego zapytania zwiększ szerokość okna terminala i uruchom poniższe polecenia celem poprawy czytelności wyniku. Jeśli korzystasz s SQL

Developera wywołaj powyższe polecenie przyciskiem 🕨

```
column versions_starttime format a20
column versions_endtime format a20
column opis format a20
set line 200
```

- Ile wersji rekordu zostało zarejestrowanych?
- Na podstawie czego możesz wskazać aktualną wersję rekordu?
- Jak długo obowiązywała wersja rekordu z wartością w kolumnie opis 'wartość druga'?
- Ile kolumn powyższego zapytania jest pseudokolumnami?
- Zapisz wartość kolumny versions\_xid dla wersji drugiej rekordu.

## 4. Flashback Transaction Query

Czas trwania: 5 minut

Mechanizm *Flashback Transaction Query* pozwala administratorowi nie tylko na zaglądanie do poprzednich wersji danego rekordu, ale także w prosty sposób powrócić do poprzednich wartości.

1. Korzystając z wartości wartość kolumny versions\_xid dla wersji drugiej rekordu odczytanej w poprzednim zadaniu wykonaj jako użytkownik administracyjny następujące polecenia.

```
column operation format a10
column logon_user format a10
SELECT xid, operation, start_scn, commit_scn,
logon_user, undo_sql
FROM flashback_transaction_query
WHERE xid = HEXTORAW('090021008F020000');
```

• Czy odczytane dane pozwolą odtworzyć postać rekordu sprzed "drugiej modyfikacji"?

2. Skopiuj polecenie wyświetlone w kolumnie undo\_sql z powyższego zapytania i jako użytkownik SCOTT wykonaj to polecenie i zatwierdź transakcje.

```
update "SCOTT"."TEST2"
set "OPIS" = 'wartość pierwsza'
where ROWID = 'AAASNQAAEAAAAIfAAA';
commit;
```

3. Sprawdź jako użytkownik SCOTT aktualną zawartość tabeli test2.

# 5. Flashback Table

Czas trwania: 10 minut

Retrospekcja może być wykorzystana do szybkiego przywrócenia zawartości poszczególnych tabel do ich stanu w przeszłości.

1. Aby ułatwić sobie dalsze ćwiczenia i nie przełączać się pomiędzy sesjami użytkownika SCOTT i administratora nadamy prawo odczytu słownika bazy danych użytkownikowi SCOTT. Uruchom poniższe polecenie jako administrator.

GRANT SELECT ANY DICTIONARY TO SCOTT;

2. Jako użytkownik SCOTT utwórz nową tabelę i włącz możliwość przenoszenia wierszy (pomiędzy segmentami tej tabeli).

CREATE TABLE test3 ( id NUMBER);

ALTER TABLE test3 ENABLE ROW MOVEMENT;

3. Sprawdź czy uprawnienia dla SCOTT są wystarczające aby mógł przeglądać zawartość perspektywy v\$database. Sprawdź bieżącą wartość numeru SCN i czasu. Wykonaj jako SCOTT poniższe polecenie.

```
SELECT current_scn,
TO_CHAR(SYSTIMESTAMP, 'YYYY-MM-DD HH24:MI:SS')
FROM v$database;
```

- Zanotuj odczytany numer SCN. Będzie to SCN "po utworzeniu tabeli".
- 4. Dodaj do nowej tabeli nowy rekord i zatwierdź transakcję.

INSERT INTO test3 VALUES (1);

COMMIT;

5. Ponownie sprawdź bieżącą wartość numeru SCN i czasu. Wykonaj poniższe polecenie.

```
SELECT current_scn,
TO_CHAR(SYSTIMESTAMP, 'YYYY-MM-DD HH24:MI:SS')
FROM v$database;
```

- Zanotuj odczytany numer SCN. Będzie to SCN "po wstawieniu rekordu".
- 6. Przewrócimy teraz zawartość tabeli do momentu "po utworzeniu tabeli". Wykonaj w tym celu jako SCOTT poniższe polecenie. Użyj właściwego numeru SCN.

FLASHBACK TABLE test3 TO SCN 842721;

7. Sprawdź efekt.

SELECT COUNT(\*) FROM test3;

8. Wróć do stanu tabeli z czasu "po wstawieniu rekordu". Użyj właściwego numeru SCN.

FLASHBACK TABLE test3 TO SCN 842921;

9. Sprawdź efekt.

SELECT COUNT(\*) FROM test3;

• Ile rekordów posiada tabela test3?

#### 6. Flashback Drop

Czas trwania: 15 minut

Zupełnie inną techniką w stosunku do tych, które do tej pory użyliśmy, jest retrospekcja usunięcia tabeli (drop). Daje ona możliwość odzyskania usuniętej tabeli.

1. Na początku jako użytkownik administracyjny sprawdzimy czy opcja ta jest włączona. W tym celu wywołaj następujące polecenie.

show parameter RECYCLEBIN

ADMINISTRATOR>show parameter RECYCLEBIN		
NAME	TYPE	
VALUE		
recyclebin on	string	=

Domyślnie opcja ta jest włączona. Gdyby jednak tak nie było, można ją włączyć poleceniem alter system set RECYCLEBIN=ON SCOPE=BOTH;

- 2. Jako użytkownik SCOTT:
  - a. utwórz nową tabelę,
  - b. dodaj do niej nowy rekord,
  - c. zatwierdź transakcję,
  - d. utwórz indeks, a następnie
  - e. utwórz wyzwalacz.

W tym celu wykonaj następujące polecenia:

```
CREATE TABLE test4 (id NUMBER);
INSERT INTO test4 VALUES (1);
COMMIT;
create index test4_indx on test4(id);
create trigger test4_trig
before delete on test4
begin
   null;
end;
/
```

3. Następnie skasujemy tabelę (wraz ze wszystkimi związanymi z nią obiektami).

DROP TABLE test4;

4. Za pomocą poniższego zapytania uzyskamy informacje, które znalazły się "w koszu" w wyniku usunięcia tabeli. Wykonaj to zapytanie jako SCOTT.

```
column object_name format a30;
column original_name format a20;
column type format a10;
SELECT OBJECT_NAME, ORIGINAL_NAME, TYPE
FROM USER_RECYCLEBIN
WHERE BASE_OBJECT in
(SELECT BASE_OBJECT
FROM USER_RECYCLEBIN
WHERE ORIGINAL_NAME = 'TEST4');
```

- Ile usuniętych obiektów jest powiązanych z tabelą test4?
- 5. Ponownie utworzymy tabelę o tej samej nazwie test4. Ponadto wstawimy do niej nowy rekord (tym razem o wartości 2) i zatwierdzimy zmiany.

```
CREATE TABLE test4 (id NUMBER);
INSERT INTO test4 VALUES (2);
COMMIT;
```

6. Następnie ponownie skasujemy tabelę.

DROP TABLE test4;

- 7. Na początku dokonamy prostego odzyskania tabeli. W tym celu:
  - a. Sprawdź jakie obiekty znajdują się w koszu. W tym celu rozwiń w *SQL Developerze* drzewo połączenia, a następnie znajdź węzeł **Recycle Bin** i rozwiń go.



Możesz również sprawdzić zawartość kosza poniższym zapytaniem

```
column object_name format a30;
column original_name format a20;
column type format a10;
SELECT OBJECT_NAME, ORIGINAL_NAME, TYPE
FROM USER_RECYCLEBIN
WHERE BASE_OBJECT in
(SELECT BASE_OBJECT
FROM USER_RECYCLEBIN
WHERE ORIGINAL_NAME = 'TEST4');
```

Zwróć uwagę, że zapytanie pokazuje dwa rekordy, opisujące usuniętą tabelę, tymczasem *SQL Developer* w węźle **Recycle Bin** pokazuje tylko jeden wpis dotyczący usuniętej tabeli.

b. Następnie wykonaj polecenie przywracające usuniętą tabelę.

FLASHBACK TABLE test4 TO BEFORE DROP;

c. Sprawdź efekt przywrócenia tabeli.

SELECT \* FROM test4;

- Która tabela została odtworzona? Pierwsza czy ostatnia?
- 8. Załóżmy, że chcieliśmy przywrócić pierwszą z tabel. W takim przypadku postępowanie będzie odrobinę bardziej złożone.
  - a. Ponownie usuniemy drugą tabelę.

DROP TABLE test4;

b. Następnie przeglądniemy zawartość kosza w poszukiwaniu "właściwej" tabeli poniższym poleceniem.

```
SELECT OBJECT_NAME, DROPTIME
FROM USER_RECYCLEBIN
WHERE ORIGINAL_NAME = 'TEST4'
AND TYPE = 'TABLE';
```

- Ile usuniętych tabel test4 zostało odczytanych?
- Zanotuj wartość OBJECT\_NAME tabeli usuniętej jako pierwszej.
- Zanotuj wartość OBJECT\_NAME tabeli usuniętej jako drugiej.
- c. Aby upewnić się odnośnie zawartości każdej z tych tabel wykonaj zapytania używając w tym celu zanotowanych wartości OBJECT\_NAME.

```
SELECT * FROM "BIN$nzzu2DPAqhrgQAB/AQBJzQ==$0";
SELECT * FROM "BIN$nzzu2DPCqhrgQAB/AQBJzQ==$0";
```

d. Znając obecną "nazwę" pierwszej tabeli możemy dokonać jej odtworzenia. W tym celu wykonaj następujące polecenie. Pamiętaj aby użyć właściwej "nazwy".

```
FLASHBACK TABLE "BIN$nzzu2DPAqhrgQAB/AQBJzQ==$0" TO BEFORE
DROP RENAME TO TEST4_1;
```

e. Sprawdź efekt przywrócenia tabeli.

SELECT \* FROM test4\_1;

- Czy tym razem udało Ci się odtworzyć pierwszą tabelę?
- f. Sprawdź czy inne obiekty powiązane z tą tabelą zostały odtworzone. Wykonaj poniższe polecenia.

```
SELECT index_name
FROM user_indexes
WHERE table_name = 'TEST4_1';
SELECT trigger_name
FROM user_triggers
WHERE table name = 'TEST4 1';
```

- Czy indeks na tej tabeli został odtworzony?
- Czy wzwalacz na tej tabeli został odtworzony?

Sprawdź dokładnie. Być może wykonaj inne dodatkowe polecenia.

g. Oczywiście nic nie stoi na przeszkodzie aby odtworzyć też tę drugą tabelę.

```
FLASHBACK TABLE "BIN$nzzu2DPCqhrgQAB/AQBJzQ==$0" TO BEFORE
DROP RENAME TO TEST4_2;
```

h. Sprawdź efekt przywrócenia tabeli.

```
SELECT * FROM test4_2;
```

9. Jeśli usuwane tabele na pewno nie będą potrzebne w przyszłości można je usunąć bez wykorzystania kosza. Usuń w ten sposób odtworzone tabele. Na zakończenie sprawdź zawartość kosza.

```
DROP TABLE test4_1 PURGE;
DROP TABLE test4_2 PURGE;
SELECT OBJECT_NAME, ORIGINAL_NAME, TYPE FROM USER_RECYCLEBIN;
```

• Czy tym razem coś po usuwanych tabelach pozostało?

# 7. Flashback Database

#### Czas trwania: 10 minut

Na zakończenie ćwiczeń z retrospekcji wykonamy retrospekcję całej bazy danych.

1. Na początku jako użytkownik SCOTT sprawdź jakie posiadasz w chwili obecnej tabele.

SELECT table\_name FROM user\_tables;

2. Następnie rozłącz sesję użytkownika SCOTT.



3. Jako administrator bazy danych zamknij bazę danych, a następnie uruchom ją w trybie MOUNT (używanym najczęściej podczas odtwarzania bazy danych).

SHUTDOWN IMMEDIATE;

STARTUP MOUNT;

```
ADMINISTRATOR>SHUTDOWN IMMEDIATE;
Baza danych została zamknięta.
Baza danych została zdemontowana.
Instancja ORACLE została zamknięta.
ADMINISTRATOR>STARTUP MOUNT;
Instancja ORACLE została uruchomiona.
Total System Global Area 849530880 bytes
Fixed Size 1339824 bytes
Variable Size 503320144 bytes
Database Buffers 339738624 bytes
Redo Buffers 5132288 bytes
Baza danych została zamontowana.
ADMINISTRATOR>
```

4. Następnie przypomnij sobie pierwszy numer SCN jaki odczytałeś podczas ćwiczeń z retrospekcji. Dokonaj odtworzenia bazy danych do tego punktu w czasie. Wykorzystaj w tym celu mechanizm retrospekcji bazy danych.

FLASHBACK DATABASE TO SCN 834824;

- Czy w chwili obecnej baza danych jest w stanie spójnym?
- 5. Spróbuj otworzyć bazę danych poniższym poleceniem.

ALTER DATABASE OPEN;

- Czy polecenie się powiodło?
- 6. Ponownie spróbuj otworzyć bazę danych. Tym razem użyj opcji RESETLOGS.

ALTER DATABASE OPEN RESETLOGS;

- Czy polecenie się powiodło?
- 7. Zaloguj się jako użytkownik SCOTT i sprawdź jakie tabele znajdują się w Twoim schemacie.

SELECT table\_name FROM user\_tables;

- Czy istnieje w schemacie użytkownika SCOTT tabela test2?
- 8. Sprawdź jako użytkownik SCOTT bieżącą wartość numeru SCN i czasu. Wykonaj poniższe polecenie.

```
SELECT current_scn,
        TO_CHAR(SYSTIMESTAMP, 'YYYY-MM-DD HH24:MI:SS')
FROM v$database;
```

- Czy polecenie się powiodło?
- Dlaczego?