Anton Slizh's

U2M5.LW.Advanced SQL, PL/SQL

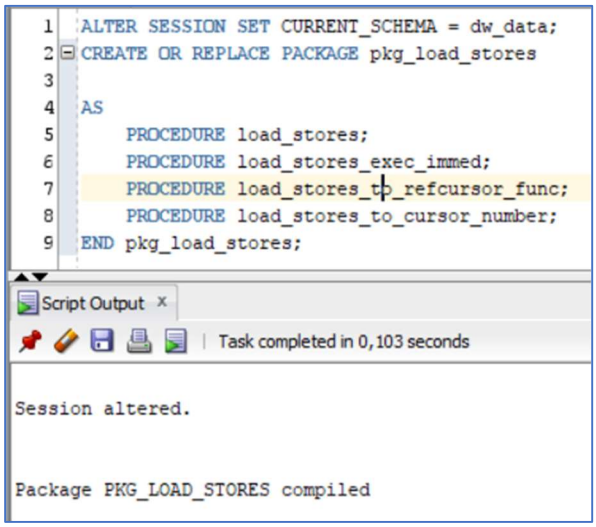
GitHub: https://github.com/drapejny/DataCamp2022

Task 1

2.1. Task 01: Create Packages for Reload Dimension from SA_*

The main task was to refactor packages from previous task using EXECUTE IMMEDIATE, TO_REFCURSOR and TO_CURSOR_NUMBER function.

I have added some additional procedures into loading store package:



All scripts are located at *pkg_load_stores_impl.sql*

1. Using EXECUTE IMMEDIATE with Bind Params

PROCEDURE load_stores_exec_immed

```
110                         cl.city,
111                         cl.phone
112             FROM dw_cl.dw_cl_store_data cl
113             LEFT JOIN dw_data.dw_store_data dw
114             ON cl.address = dw.address;
115
116         FETCH c_store
117
118         BULK COLLECT INTO new_stores;
119
120         CLOSE c_store;
121
122         sql_insert_stmt := 'INSERT INTO dw_data.dw_store_data VALUES( :1, :2, :3, :4, :5, :6)';
123         sql_update_stmt := 'UPDATE dw_data.dw_store_data SET address = :1, country = :2, region = :3, city = :4, phone = :5 WHERE store_id = :6';
124
125         FOR i IN 1 .. new_stores.COUNT LOOP
126             IF new_stores(i).store_id IS NULL THEN
127                 EXECUTE IMMEDIATE sql_insert_stmt
128                 USING dw_data.seq_stores.NEXTVAL, new_stores(i).address, new_stores(i).country, new_stores(i).region, new_stores(i).city, new_stores(i).phone;
129             ELSE
130                 EXECUTE IMMEDIATE sql_update_stmt
131                 USING new_stores(i).address, new_stores(i).country, new_stores(i).region, new_stores(i).city, new_stores(i).phone, new_stores(i).store_id;
132             END IF;
133         END LOOP;
134
135         COMMIT;
136
137     END load_stores_exec_immed;
```

Executing:

```
1  begin
2      dw_data.pkg_load_stores.load_stores_exec_immed;
3  end;
4  select * from dw_data.dw_store_data;
```

Script Output | Query Result | Task completed in 0,039 seconds

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

Result:

```
4  select * from dw_data.dw_store_data;
```

Script Output | Query Result | SQL | All Rows Fetched: 16 in 0,027 seconds

	STORE_ID	ADDRESS	COUNTRY	REGION	CITY	PHONE
1	170	Gomel, Kalinouskaga, 6	Belarus	Gomel	Gomel	+375291110892
2	171	Minks, Byady, 3	Belarus	Minsk	Minsk	+375291122892
3	172	Astana, Kalinouskaga, 90	Kazakhstan	Astana	Astana	+99677131211
4	173	Novogrudok, Kalinouskaga, 1	Belarus	Grodno	Novogrudok	+375788822892
5	174	Mogilev, Kalinouskaga, 17	Belarus	Mogilev	Mogilev	+375290121234
6	175	Minks, Kalinouskaga, 33	Belarus	Minsk	Minsk	+375291122892
7	176	Grodno, Savetskaya, 2	Belarus	Grodno	Grodno	+375291122552
8	177	Grodno, Kirova, 13	Belarus	Grodno	Grodno	+375294322892
9	178	Brest, Kalinouskaga, 47	Belarus	Brest	Brest	+375290120092
10	179	Kyiv, Kalinouskaga, 11	Ukraine	Kyiv	Kyiv	+380291144894
11	180	Minks, Zaporozhskaya, 73	Belarus	Minsk	Minsk	+375291522852
12	181	Grodno, Kalinouskaga, 7	Belarus	Grodno	Grodno	+375290122892
13	182	Moscow, Kalinouskaga, 12	Russian Federation	Moscow	Moscow	+71232131232
14	183	Saint Petersburg, Kalinouskaga, 90	Russian Federation	Leningrad	Saint Petersburg	+71277131232
15	184	Vitebsk, Kalinouskaga, 9	Belarus	Vitebsk	Vitebsk	+375291111892
16	185	Slonim, Skaryny, 18	Belarus	Grodno	Slonim	+375290292892

2. Using DBMS_SQL.TO_REFCURSOR Function

PROCEDURE load_stores_to_refcursor_func

```
161             ON cl.address = dw.address';
162
163         cursor_id := DBMS_SQL.open_cursor;
164
165         DBMS_SQL.PARSE(cursor_id, query_text, DBMS_SQL.NATIVE);
166
167         cur_count := DBMS_SQL.EXECUTE(cursor_id);
168
169         ref_cursor := DBMS_SQL.TO_REFCURSOR(cursor_id);
170
171         LOOP
172         FETCH ref_cursor INTO store_row;
173             EXIT WHEN ref_cursor%NOTFOUND;
174             IF store_row.store_id IS NULL THEN
175              INSERT INTO dw_data.dw_store_data
176                     VALUES (
177                             dw_data.seq_stores.NEXTVAL,
178                             store_row.address,
179                             store_row.country,
180                             store_row.region,
181                             store_row.city,
182                             store_row.phone
183                            );
184             ELSE
185                 UPDATE dw_data.dw_store_data
186                 SET address = store_row.address,
187                     country = store_row.country,
188                     region = store_row.region,
189                     city = store_row.city,
```

Executing:

```
2  begin
3      dw_data.pkg_load_stores.load_stores_to_refcursor_func;
4  end;
```

Script Output × | Query Result ×

Task completed in 0,103 seconds

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

Result:

```
5  select * from dw_data.dw_store_data;
```

Script Output × | Query Result ×

SQL | All Rows Fetched: 16 in 0,028 seconds

	STORE_ID	ADDRESS	COUNTRY	REGION	CITY	PHONE
1	186	Gomel, Kalinouskaga, 6	Belarus	Gomel	Gomel	+375291110892
2	187	Minks, Byady, 3	Belarus	Minsk	Minsk	+375291122892
3	188	Astana, Kalinouskaga, 90	Kazakhstan	Astana	Astana	+99677131211
4	189	Novogrudok, Kalinouskaga, 1	Belarus	Grodno	Novogrudok	+375788822892
5	190	Mogilev, Kalinouskaga, 17	Belarus	Mogilev	Mogilev	+375290121234
6	191	Minks, Kalinouskaga, 33	Belarus	Minsk	Minsk	+375291122892
7	192	Grodno, Savetskaya, 2	Belarus	Grodno	Grodno	+375291122552
8	193	Grodno, Kirova, 13	Belarus	Grodno	Grodno	+375294322892
9	194	Brest, Kalinouskaga, 47	Belarus	Brest	Brest	+375290120092
10	195	Kyiv, Kalinouskaga, 11	Ukraine	Kyiv	Kyiv	+380291144894
11	196	Minks, Zaporozhskaya, 73	Belarus	Minsk	Minsk	+375291522852
12	197	Grodno, Kalinouskaga, 7	Belarus	Grodno	Grodno	+375290122892
13	198	Moscow, Kalinouskaga, 12	Russian Federation	Moscow	Moscow	+71232131232
14	199	Saint Petersburg, Kalinouskaga, 90	Russian Federation	Leningrad	Saint Petersburg	+71277131232
15	200	Vitebsk, Kalinouskaga, 9	Belarus	Vitebsk	Vitebsk	+375291111892
16	201	Slonim, Skaryny, 18	Belarus	Grodno	Slonim	+375290292892

3. Using DBMS_SQL.TO_CURSOR_NUMBER Function

PROCEDURE load_stores_to_cursor_number

```
224             ON cl.address = dw.address';
225
226         OPEN src_cur FOR query_text;
227
228         curid := DBMS_SQL.TO_CURSOR_NUMBER(src_cur);
229
230         DBMS_SQL.describe_columns(curid, colcnt, desctab);
231
232         FOR i IN 1 .. colcnt
233         LOOP
234             CASE desctab(i).col_type
235                 WHEN 1 THEN DBMS_SQL.define_column (curid, i, varchar2_val, 4000);
236                 WHEN 2 THEN DBMS_SQL.define_column (curid, i, number_val);
237                 ELSE DBMS_SQL.define_column (curid, i, varchar2_val, 4000);
238             END CASE;
239
240         END LOOP;
241
242         WHILE DBMS_SQL.FETCH_ROWS(curid) > 0
243         LOOP
244             FOR i IN 1 .. colcnt
245             LOOP
246                 CASE desctab(i).col_type
247                     WHEN 1 THEN
248                         DBMS_SQL.COLUMN_VALUE (curid, i, varchar2_val);
249                         CASE desctab(i).col_name
250                             WHEN 'ADDRESS' THEN store_row.address := varchar2_val;
251                             WHEN 'COUNTRY' THEN store_row.country := varchar2_val;
252                             WHEN 'REGION' THEN store_row.region := varchar2_val;
```

Executing:

```
2  begin
3      dw_data.pkg_load_stores.load_stores_to_cursor_number;
4  end;
```

Script Output | Query Result

Task completed in 0,046 seconds

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

Result:

```
5  select * from dw_data.dw_store_data;
```

Script Output | Query Result

SQL | All Rows Fetched: 16 in 0,029 seconds

	STORE_ID	ADDRESS	COUNTRY	REGION	CITY	PHONE
1	202	Gomel, Kalinouskaga, 6	Belarus	Gomel	Gomel	+375291110892
2	203	Minks, Byady, 3	Belarus	Minsk	Minsk	+375291122892
3	204	Astana, Kalinouskaga, 90	Kazakhstan	Astana	Astana	+99677131211
4	205	Novogrudok, Kalinouskaga, 1	Belarus	Grodno	Novogrudok	+375788822892
5	206	Mogilev, Kalinouskaga, 17	Belarus	Mogilev	Mogilev	+375290121234
6	207	Minks, Kalinouskaga, 33	Belarus	Minsk	Minsk	+375291122892
7	208	Grodno, Savetskaya, 2	Belarus	Grodno	Grodno	+375291122552
8	209	Grodno, Kirova, 13	Belarus	Grodno	Grodno	+375294322892
9	210	Brest, Kalinouskaga, 47	Belarus	Brest	Brest	+375290120092
10	211	Kyiv, Kalinouskaga, 11	Ukraine	Kyiv	Kyiv	+380291144894
11	212	Minks, Zaporozhskaya, 73	Belarus	Minsk	Minsk	+375291522852
12	213	Grodno, Kalinouskaga, 7	Belarus	Grodno	Grodno	+375290122892
13	214	Moscow, Kalinouskaga, 12	Russian Federation	Moscow	Moscow	+71232131232
14	215	Saint Petersburg, Kalinouskaga, 90	Russian Federation	Leningrad	Saint Petersburg	+71277131232
15	216	Vitebsk, Kalinouskaga, 9	Belarus	Vitebsk	Vitebsk	+375291111892
16	217	Slonim, Skaryny, 18	Belarus	Grodno	Slonim	+375290292892

3.1. Task 02: CREATE Monthly Reports Layouts

Let’s calculate monthly sales report using Model Clause

```
WITH sales_by_month
AS
(
    SELECT TRUNC(date_id, 'MM') AS month,
           product_name AS product,
           SUM(amount * price) AS revenue,
           sum(amount) AS amount
            FROM sa_customers.sa_sale_data s
           JOIN sa_products.sa_product_data p
       ON s.sku_num = p.sku_num
       GROUP BY TRUNC(date_id, 'MM'), product_name
)

SELECT DISTINCT month, product, amount, revenue
FROM sales_by_month

MODEL
    PARTITION BY (month)
    DIMENSION BY (product)
    MEASURES (revenue, amount)
    RULES
        (
            revenue['All products'] = SUM(revenue)[any],
            amount['All products'] = SUM(amount)[any]
        )
ORDER BY month, revenue DESC;
```

Result:

	MONTH	PRODUCT	AMOUNT	REVENUE
1	01.01.21	All products	32448	69068,1
2	01.01.21	Zaporozhski Light 2 liter	392	1960
3	01.01.21	Zaporozhski Dark 2 liter	370	1665
4	01.01.21	Zaporozhski Classic 2 liter	396	1584
5	01.01.21	Zaporozhski Dark 1.5 liter	442	1547
6	01.01.21	Zaporozhski Classic 1.5 liter	488	1464
7	01.01.21	Zaporozhski Light 1 liter	463	1389
8	01.01.21	Zaporozhski Light 0.5 liter	409	1267,9
9	01.01.21	Hatni Dark 2 liter	422	1266
10	01.01.21	Hatni Classic 2 liter	416	1248
11	01.01.21	Alivarski Dark 2 liter	416	1248
12	01.01.21	Zaporozhski Light 1.5 liter	416	1248
13	01.01.21	Alivarski Light 2 liter	429	1244,1
14	01.01.21	Lidski Light 2 liter	431	1206,8
15	01.01.21	Hatni Light 2 liter	399	1157,1
16	01.01.21	Hatni Cranberry 2 liter	444	1110
17	01.01.21	Zaporozhski Dark 0.5 liter	382	1107,8
18	01.01.21	Alivarski Classic 2 liter	362	1086
19	01.01.21	Lidski Cinnamon 2 liter	426	1065
20	01.01.21	Lidski Orange 2 liter	418	1045
21	01.01.21	Lidski Dark 2 liter	373	1044,4
22	01.01.21	Hatni Orange 2 liter	413	1032,5
23	01.01.21	Zaporozhski Classic 0.5 liter	406	1015
24	01.01.21	Lidski Classic 2 liter	377	942,5