RIGA TECHNICAL UNIVERSITY
FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
INSTITUTE OF APPLIED COMPUTER SYSTEMS

Homework #3

“Large Databases”

Use of object methods for data extraction from ORDB data storage structures. Java language use in database server.

Author: Oleg Tsoy
Course, group: ADBD0
Student card no.: 131ADB042

2014 / 2015 study year

Content

1 Goal .... **Error! Bookmark not defined.**
2 Task .... **Error! Bookmark not defined.**
3 Database description.. **Error! Bookmark not defined.**
4 SQL queries 4
5 Conclusions .... **Error! Bookmark not defined.**
6 References .... **Error! Bookmark not defined.**

1 Goal

Learn more about usage of object methods for data extraction from ORDB data storage structures. How we can use Java programming language in the database server.

2 Task

1. Creation and test of **MAP** type method.

2. Creation and test of **ORDER** type method.

3. Creation and test of **MEMBER** type method.

4. Creation and test of user-defined constructor method.

5. Java class creation and loading in database:

   1) using LOADJAVA program;

   2) using CREATE JAVA command.

Java class meta-date inspection (SELECT).

Creation of PL/SQL function which call Java method. Execution of Java method.

6. Java class creation, loading and testing for SELECT query realization.

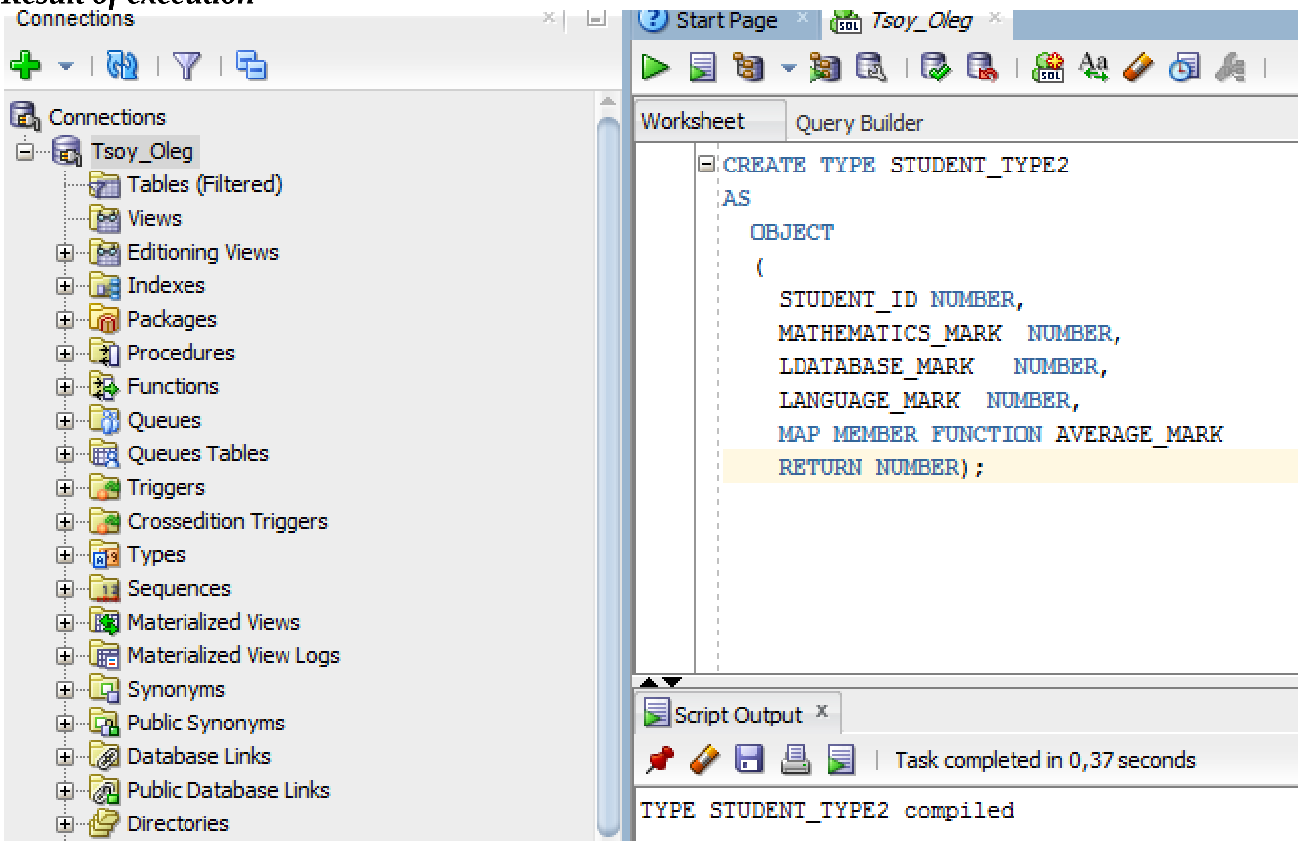
7. Conclusions (what seems good, what bad, what like, what is problematic).

3 SQL queries

1. Query goal (MAP type method):
Query SQL code

```
CREATE TYPE STUDENT_TYPE2
AS
 OBJECT
 (
  STUDENT_ID NUMBER,
  MATHEMATICS_MARK  NUMBER,
  LDATABASE_MARK   NUMBER,
  LANGUAGE_MARK  NUMBER,
  MAP MEMBER FUNCTION AVERAGE_MARK
  RETURN NUMBER);
```

Result of execution



```
CREATE TYPE STUDENT_TYPE2
AS
  OBJECT
  (
    STUDENT_ID NUMBER,
    MATHEMATICS_MARK  NUMBER,
    LDATABASE_MARK   NUMBER,
    LANGUAGE_MARK  NUMBER,
    MAP MEMBER FUNCTION AVERAGE_MARK
    RETURN NUMBER);
```

Task completed in 0,37 seconds

```
TYPE STUDENT_TYPE2 compiled
```

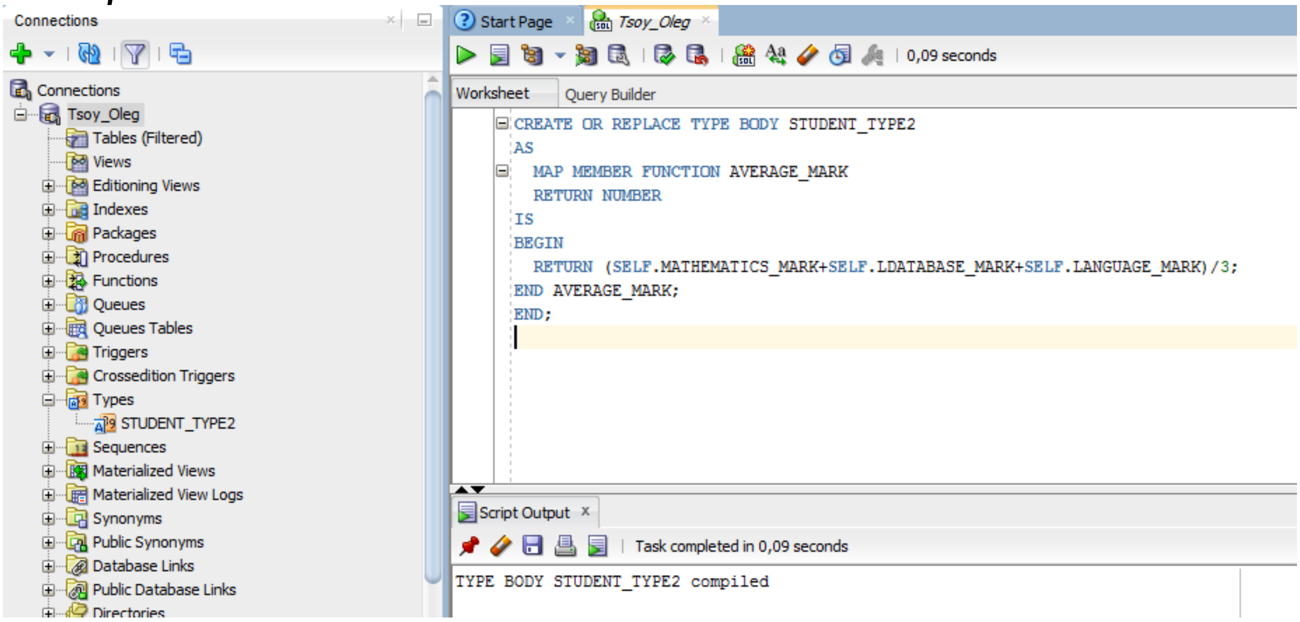
Analysis of results, what in these data can be seen: *As can be seen from the screenshot I have created the STUDENT_TYPE2 as an object for my project.*

2. Query goal (MAP type method):
Query SQL code

```
CREATE OR REPLACE TYPE BODY STUDENT_TYPE2
AS
 MAP MEMBER FUNCTION AVERAGE_MARK
 RETURN NUMBER
IS
```

BEGIN
RETURN
(SELF.MATHEMATICS_MARK+SELF.LDATABASE_MARK+SELF.LANGUAGE_MARK)/3;
END AVERAGE_MARK;
END;

Result of execution



```
CREATE OR REPLACE TYPE BODY STUDENT_TYPE2
AS
  MAP MEMBER FUNCTION AVERAGE_MARK
  RETURN NUMBER
IS
BEGIN
  RETURN (SELF.MATHEMATICS_MARK+SELF.LDATABASE_MARK+SELF.LANGUAGE_MARK)/3;
END AVERAGE_MARK;
END;
```

Task completed in 0,09 seconds

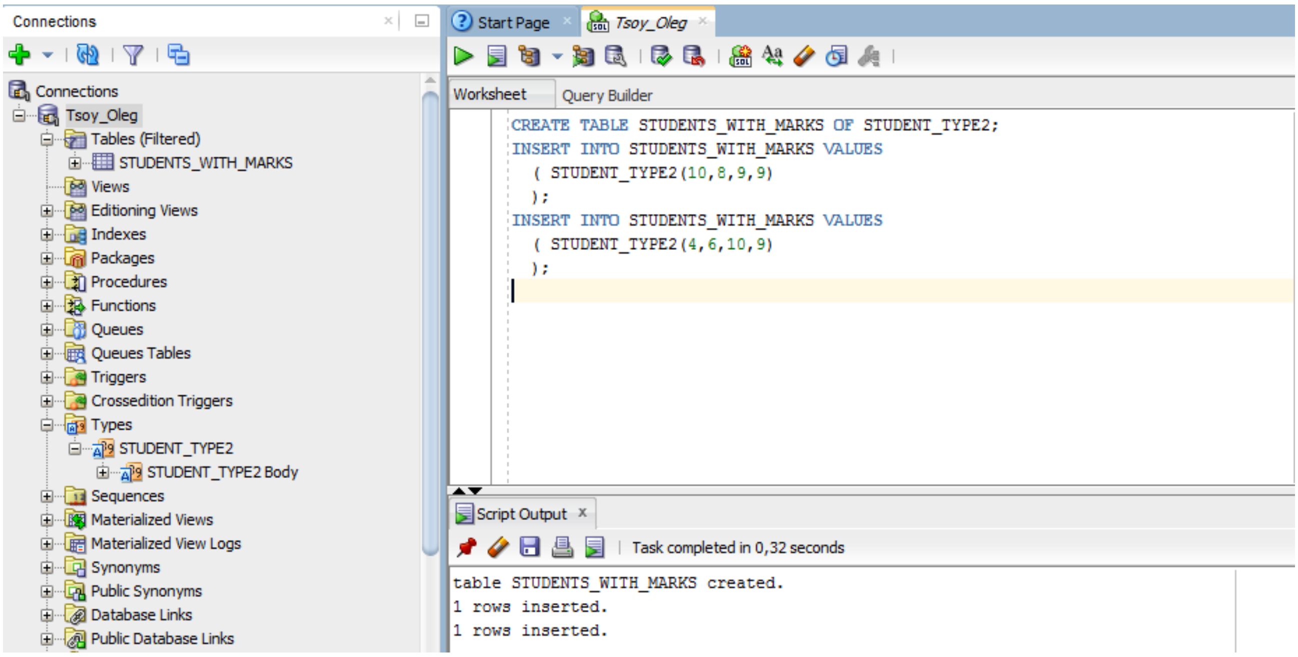
```
TYPE BODY STUDENT_TYPE2 compiled
```

Analysis of results, what in these data can be seen: *In this case I have compiled STUDENT_TYPE2 with MAP MEMBER FUNCTION.*

3. Query goal (MAP type method):
Query SQL code

CREATE TABLE STUDENTS_WITH_MARKS OF STUDENT_TYPE2;
INSERT INTO STUDENTS_WITH_MARKS VALUES
(STUDENT_TYPE2(10,8,9,9)
);
INSERT INTO STUDENTS_WITH_MARKS VALUES
(STUDENT_TYPE2(4,6,10,9)
);

Result of execution



```
CREATE TABLE STUDENTS_WITH_MARKS OF STUDENT_TYPE2;
INSERT INTO STUDENTS_WITH_MARKS VALUES
  ( STUDENT_TYPE2(10,8,9,9)
  );
INSERT INTO STUDENTS_WITH_MARKS VALUES
  ( STUDENT_TYPE2(4,6,10,9)
  );
```

Task completed in 0,32 seconds

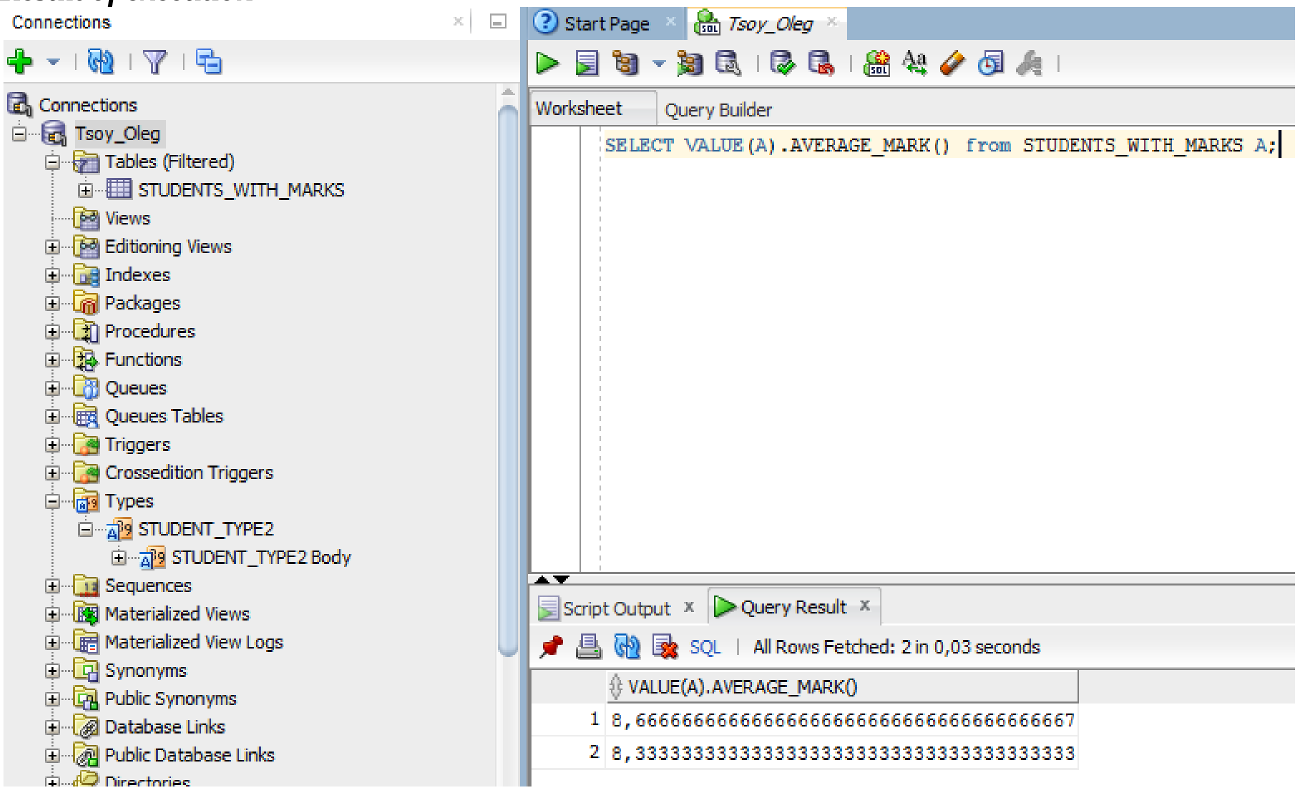
```
table STUDENTS_WITH_MARKS created.
1 rows inserted.
1 rows inserted.
```

Analysis of results, what in these data can be seen: *I have created STUDENT_WITH_MARKS table and 2 rows of data were inserted into this table.*

4. Query goal (MAP type method):
Query SQL code

SELECT VALUE(A).AVERAGE_MARK() from STUDENTS_WITH_MARKS A;

Result of execution



```
SELECT VALUE(A).AVERAGE_MARK() from STUDENTS_WITH_MARKS A;
```

All Rows Fetched: 2 in 0,03 seconds

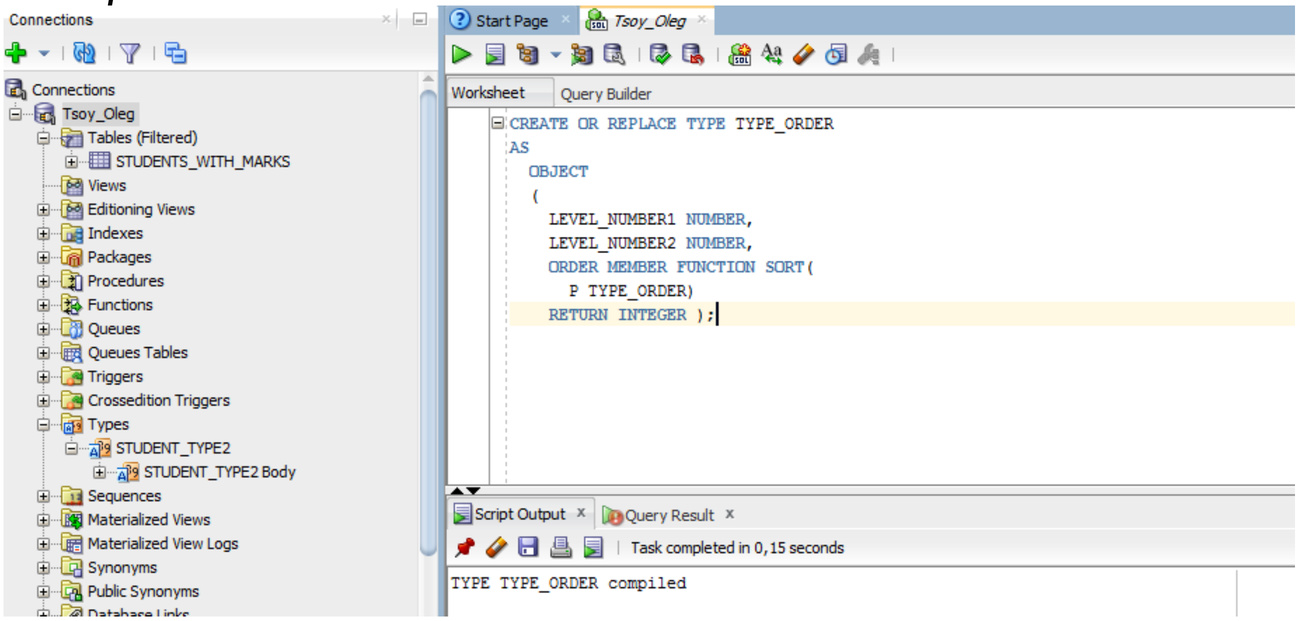
```
   VALUE(A).AVERAGE_MARK()
1  8,6666666666666666666666666666666666667
2  8,3333333333333333333333333333333333333
```

Analysis of results, what in these data can be seen: *By this select statement function I presented the calculation function which was given in previous type body state. As can be seen from screenshot it gave us the AVERAGE_MARK of inputed marks of 2 students and automatically calculate them.*

5. Query goal (ORDER type method):
Query SQL code

```
CREATE OR REPLACE TYPE TYPE_ORDER
AS
  OBJECT
  (
    LEVEL_NUMBER1 NUMBER,
    LEVEL_NUMBER2 NUMBER,
    ORDER MEMBER FUNCTION SORT(
      P TYPE_ORDER)
    RETURN INTEGER );
```

Result of execution



```
CREATE OR REPLACE TYPE TYPE_ORDER
AS
  OBJECT
  (
    LEVEL_NUMBER1 NUMBER,
    LEVEL_NUMBER2 NUMBER,
    ORDER MEMBER FUNCTION SORT(
      P TYPE_ORDER)
    RETURN INTEGER );
```

Task completed in 0,15 seconds

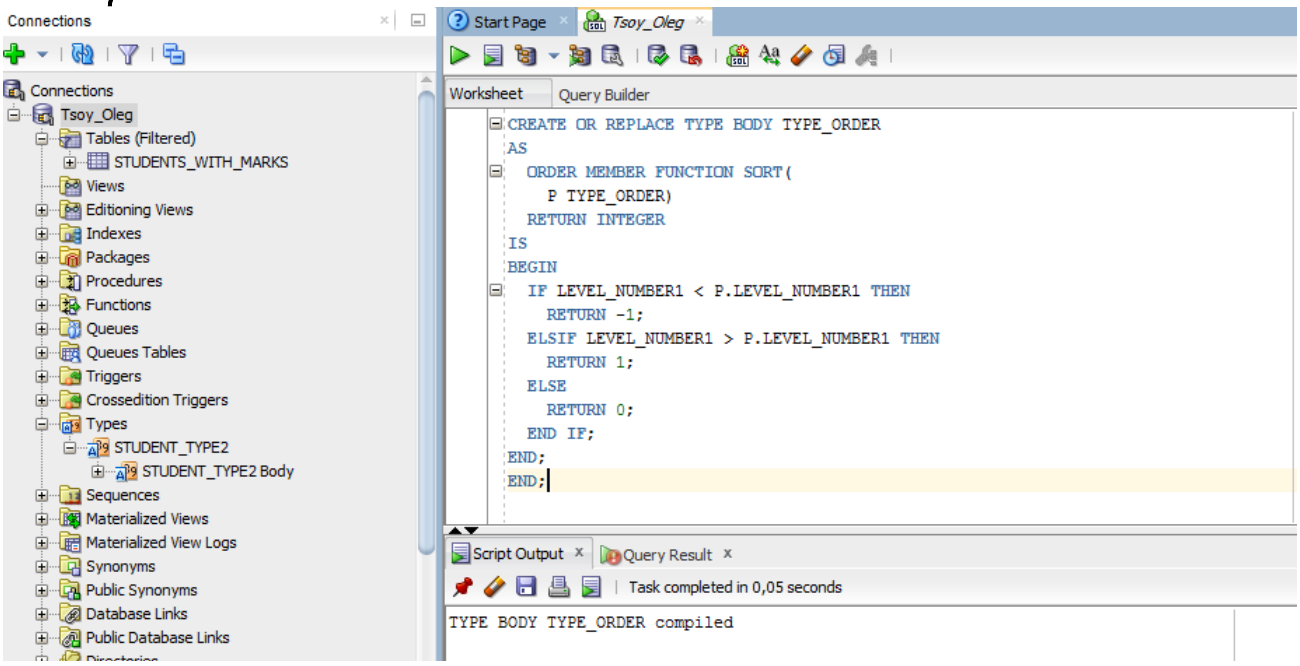
```
TYPE TYPE_ORDER compiled
```

Analysis of results, what in these data can be seen: *According to the query goal I have created TYPE_ORDER as an object to realize the ORDER function in my project.*

6. Query goal (ORDER type method):
Query SQL code

```
CREATE OR REPLACE TYPE BODY TYPE_ORDER
AS
  ORDER MEMBER FUNCTION SORT(
    P TYPE_ORDER)
  RETURN INTEGER
IS
BEGIN
  IF LEVEL_NUMBER1 < P.LEVEL_NUMBER1 THEN
    RETURN -1;
  ELSIF LEVEL_NUMBER1 > P.LEVEL_NUMBER1 THEN
    RETURN 1;
  ELSE
    RETURN 0;
  END IF;
END;
END;
```

Result of execution



```
CREATE OR REPLACE TYPE BODY TYPE_ORDER
AS
  ORDER MEMBER FUNCTION SORT(
     P TYPE_ORDER)
  RETURN INTEGER
IS
BEGIN
  IF LEVEL_NUMBER1 < P.LEVEL_NUMBER1 THEN
    RETURN -1;
  ELSIF LEVEL_NUMBER1 > P.LEVEL_NUMBER1 THEN
    RETURN 1;
  ELSE
    RETURN 0;
  END IF;
END;
END;
```

```
TYPE BODY TYPE_ORDER compiled
```

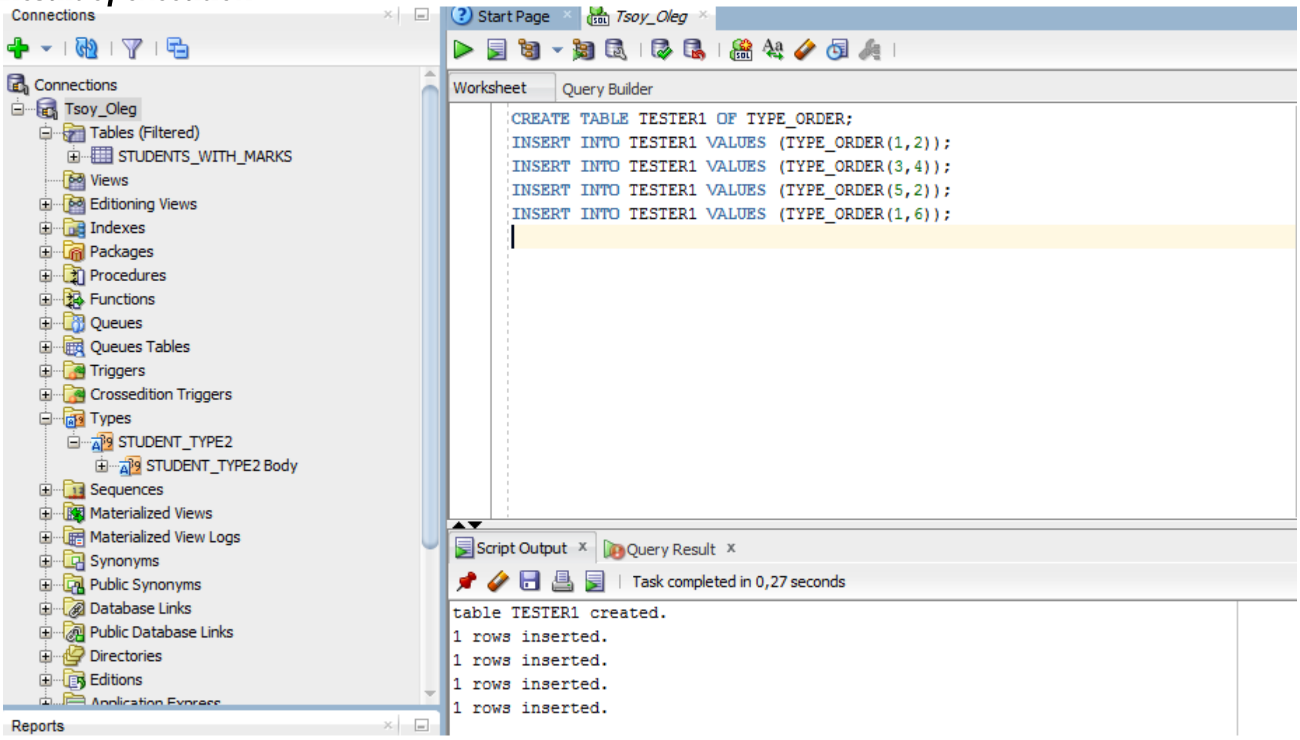
Analysis of results, what in these data can be seen: *In this case I have created the TYPE_ORDER body as order member function sort where some requirements were included.*

7. Query goal (ORDER type method):

Query SQL code

CREATE TABLE TESTER1 OF TYPE_ORDER;
INSERT INTO TESTER1 VALUES (TYPE_ORDER(1,2));
INSERT INTO TESTER1 VALUES (TYPE_ORDER(3,4));
INSERT INTO TESTER1 VALUES (TYPE_ORDER(5,2));
INSERT INTO TESTER1 VALUES (TYPE_ORDER(1,6));

Result of execution



```
CREATE TABLE TESTER1 OF TYPE_ORDER;
INSERT INTO TESTER1 VALUES (TYPE_ORDER(1,2));
INSERT INTO TESTER1 VALUES (TYPE_ORDER(3,4));
INSERT INTO TESTER1 VALUES (TYPE_ORDER(5,2));
INSERT INTO TESTER1 VALUES (TYPE_ORDER(1,6));
```

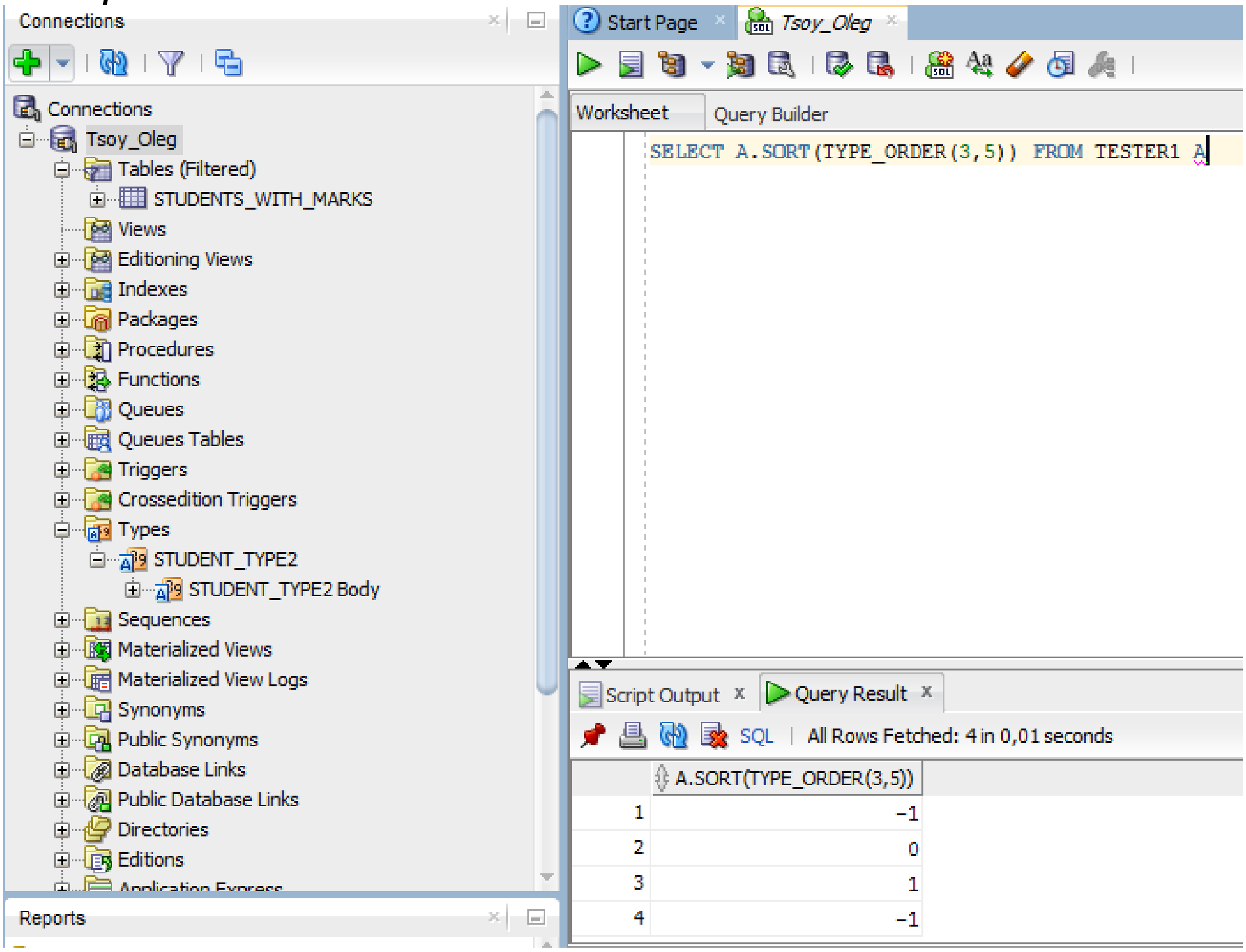
```
table TESTER1 created.
1 rows inserted.
1 rows inserted.
1 rows inserted.
1 rows inserted.
```

Analysis of results, what in these data can be seen: *In this query I have created table TESTER1 of TYPE_ORDER, where I inserted 4 rows with needed values to show the main performance of this project in ORDER MEMBER function.*

8. Query goal (ORDER type method):
Query SQL code

SELECT A.SORT(TYPE_ORDER(3,5)) FROM TESTER1 A

Result of execution



```
SELECT A.SORT(TYPE_ORDER(3,5)) FROM TESTER1 A
```

All Rows Fetched: 4 in 0,01 seconds

	A.SORT(TYPE_ORDER(3,5))
1	-1
2	0
3	1
4	-1

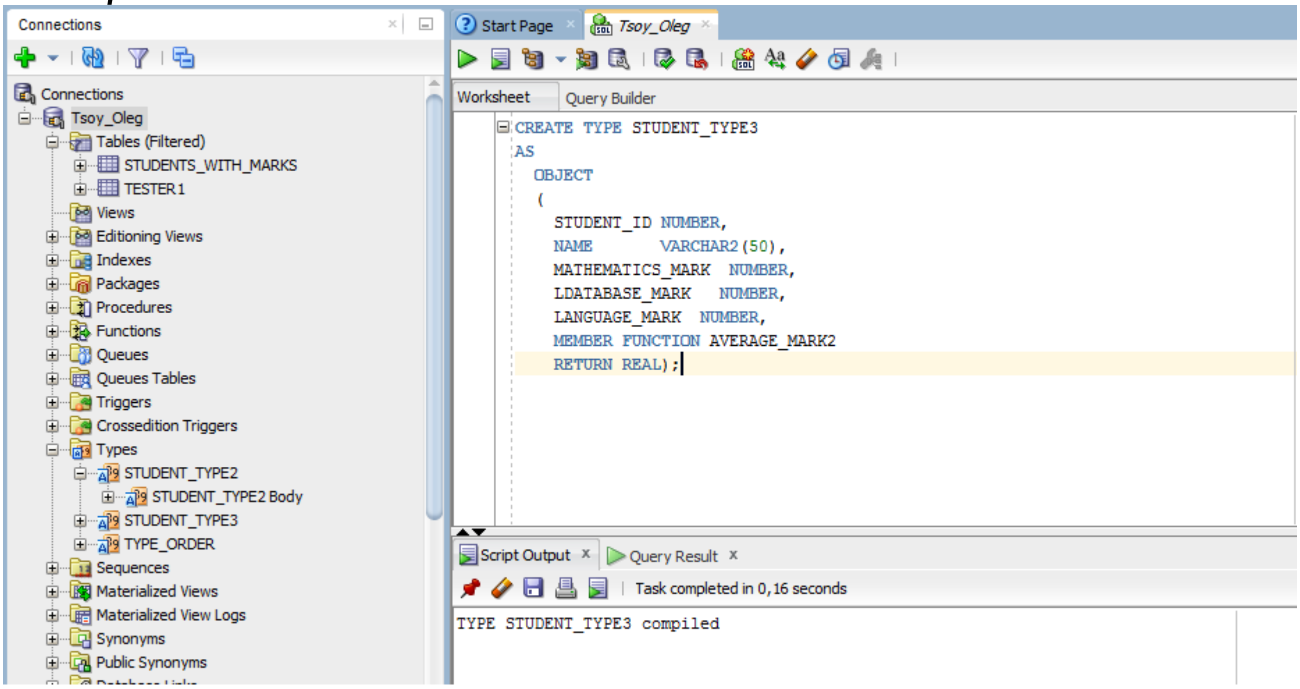
Analysis of results, what in these data can be seen: *The SELECT statement in this case shows the sort of TYPE_ORDER values which were inserted below FROM table TESTER1.*

9. Query goal (MEMBER type method):
Query SQL code

```
CREATE TYPE STUDENT_TYPE3
AS
 OBJECT
 (
  STUDENT_ID NUMBER,
  NAME       VARCHAR2(50),
  MATHEMATICS_MARK  NUMBER,
  LDATABASE_MARK   NUMBER,
  LANGUAGE_MARK  NUMBER,
  MEMBER FUNCTION AVERAGE_MARK2
```

RETURN REAL);

Result of execution



```
CREATE TYPE STUDENT_TYPE3
AS
  OBJECT
  (
    STUDENT_ID NUMBER,
    NAME          VARCHAR2(50),
    MATHEMATICS_MARK  NUMBER,
    LDATABASE_MARK    NUMBER,
    LANGUAGE_MARK  NUMBER,
    MEMBER FUNCTION AVERAGE_MARK2
    RETURN REAL);
```

Task completed in 0,16 seconds

```
TYPE STUDENT_TYPE3 compiled
```

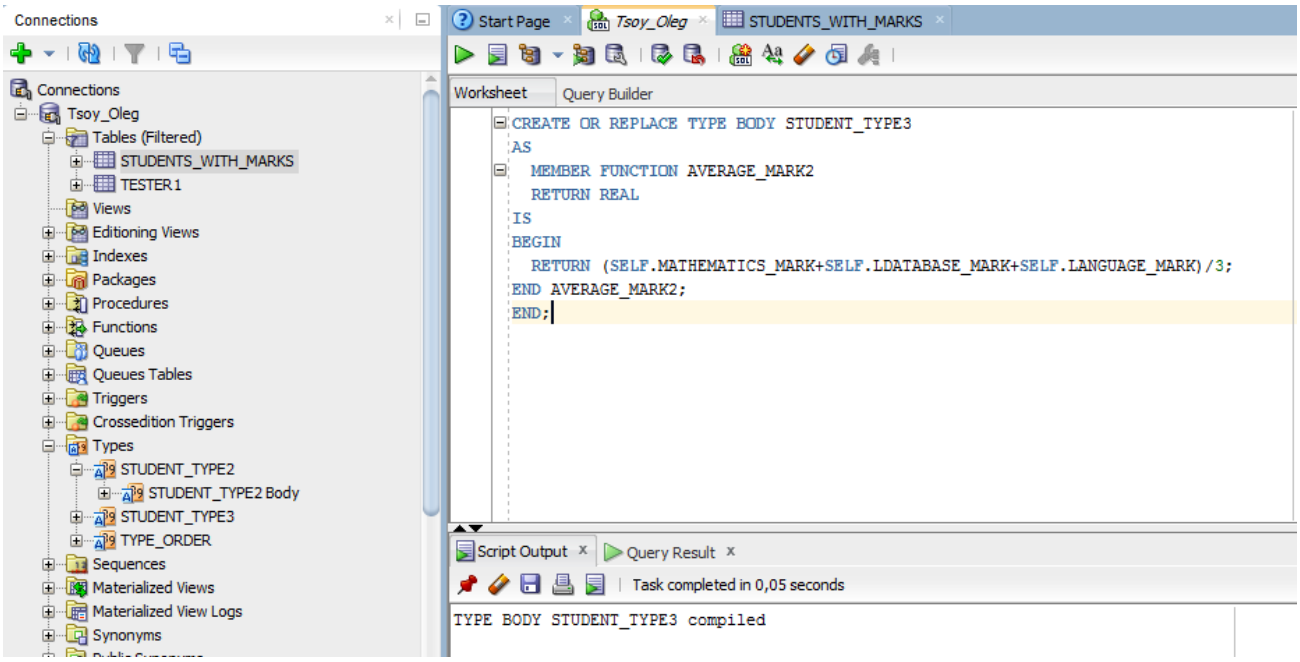
Analysis of results, what in these data can be seen: *I have created the 3rd type student as an object with ID, Name, Marks for subjects and use member function AVERAGE_MARK2 which returned real numbers.*

10. Query goal (MEMBER type method):

Query SQL code

```
CREATE OR REPLACE TYPE BODY STUDENT_TYPE3
AS
  MEMBER FUNCTION AVERAGE_MARK2
  RETURN REAL
IS
BEGIN
  RETURN
(SELF.MATHEMATICS_MARK+SELF.LDATABASE_MARK+SELF.LANGUAGE_MARK)
/3;
END AVERAGE_MARK2;
END;
```

Result of execution



```sql
CREATE OR REPLACE TYPE BODY STUDENT_TYPE3
AS
  MEMBER FUNCTION AVERAGE_MARK2
  RETURN REAL
IS
BEGIN
  RETURN (SELF.MATHEMATICS_MARK+SELF.LDATABASE_MARK+SELF.LANGUAGE_MARK)/3;
END AVERAGE_MARK2;
END;
```

Task completed in 0,05 seconds

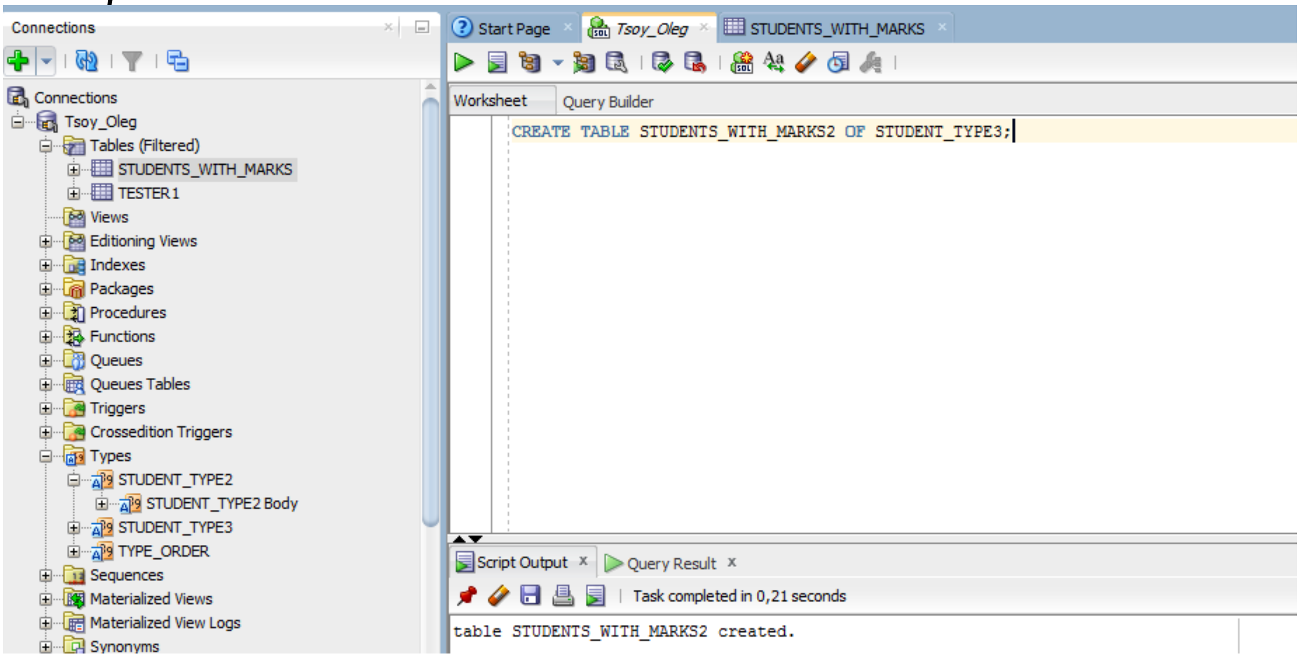
```
TYPE BODY STUDENT_TYPE3 compiled
```

Analysis of results, what in these data can be seen: *In this case the Body of student_type 3 was created with registered member function AVERAGE_MARK2, also the algorithm of calculating the AVERAGE_MARK2 was given in the code.*

11. Query goal (MEMBER type method):
Query SQL code

CREATE TABLE STUDENTS_WITH_MARKS2 OF STUDENT_TYPE3;

Result of execution



```sql
CREATE TABLE STUDENTS_WITH_MARKS2 OF STUDENT_TYPE3;
```

Task completed in 0,21 seconds

```
table STUDENTS_WITH_MARKS2 created.
```

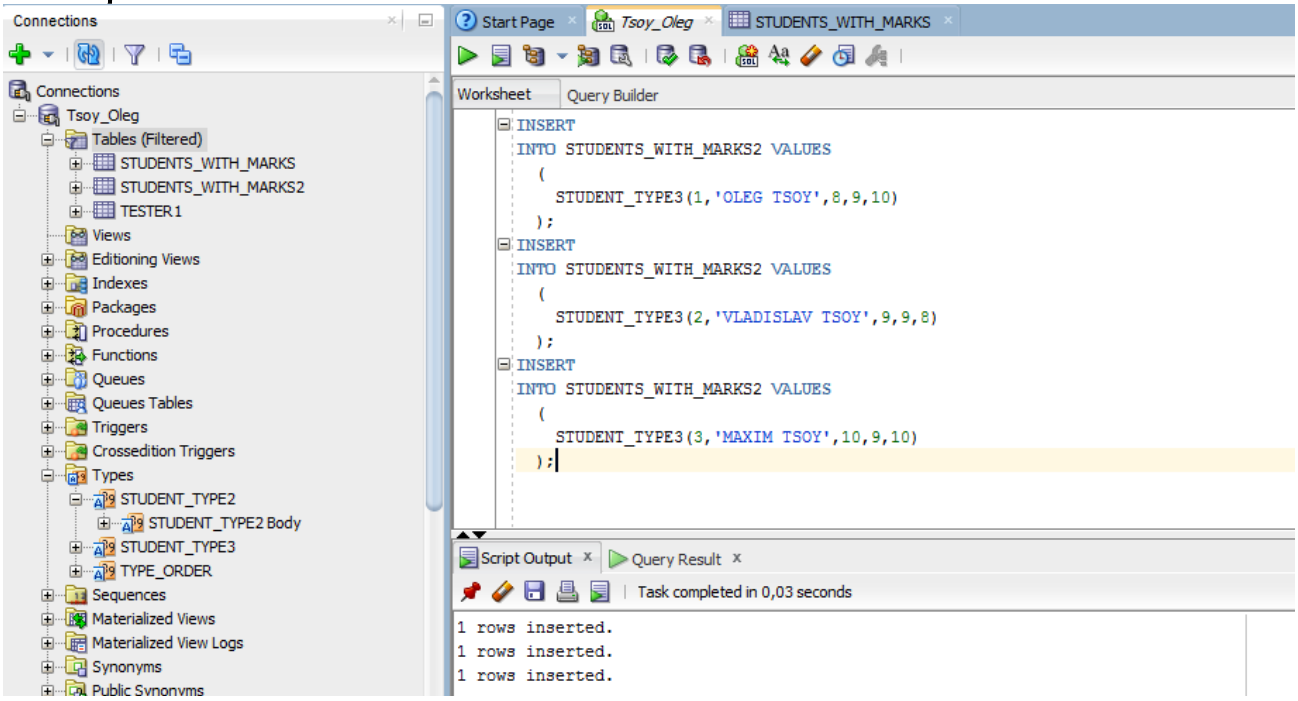
Analysis of results, what in these data can be seen: *The table STUDENTS_WITH_MARKS2 was created for STUDENT_TYPE3 observations.*

12. Query goal (MEMBER type method):
Query SQL code

INSERT

```
INTO STUDENTS_WITH_MARKS2 VALUES
  (
    STUDENT_TYPE3(1,'OLEG TSOY',8,9,10)
  );
INSERT
INTO STUDENTS_WITH_MARKS2 VALUES
  (
    STUDENT_TYPE3(2,'VLADISLAV TSOY',9,9,8)
  );
INSERT
INTO STUDENTS_WITH_MARKS2 VALUES
  (
    STUDENT_TYPE3(3,'MAXIM TSOY',10,9,10)
  );
```

Result of execution



```
INSERT
INTO STUDENTS_WITH_MARKS2 VALUES
  (
    STUDENT_TYPE3(1,'OLEG TSOY',8,9,10)
  );
INSERT
INTO STUDENTS_WITH_MARKS2 VALUES
  (
    STUDENT_TYPE3(2,'VLADISLAV TSOY',9,9,8)
  );
INSERT
INTO STUDENTS_WITH_MARKS2 VALUES
  (
    STUDENT_TYPE3(3,'MAXIM TSOY',10,9,10)
  );
```

Task completed in 0,03 seconds

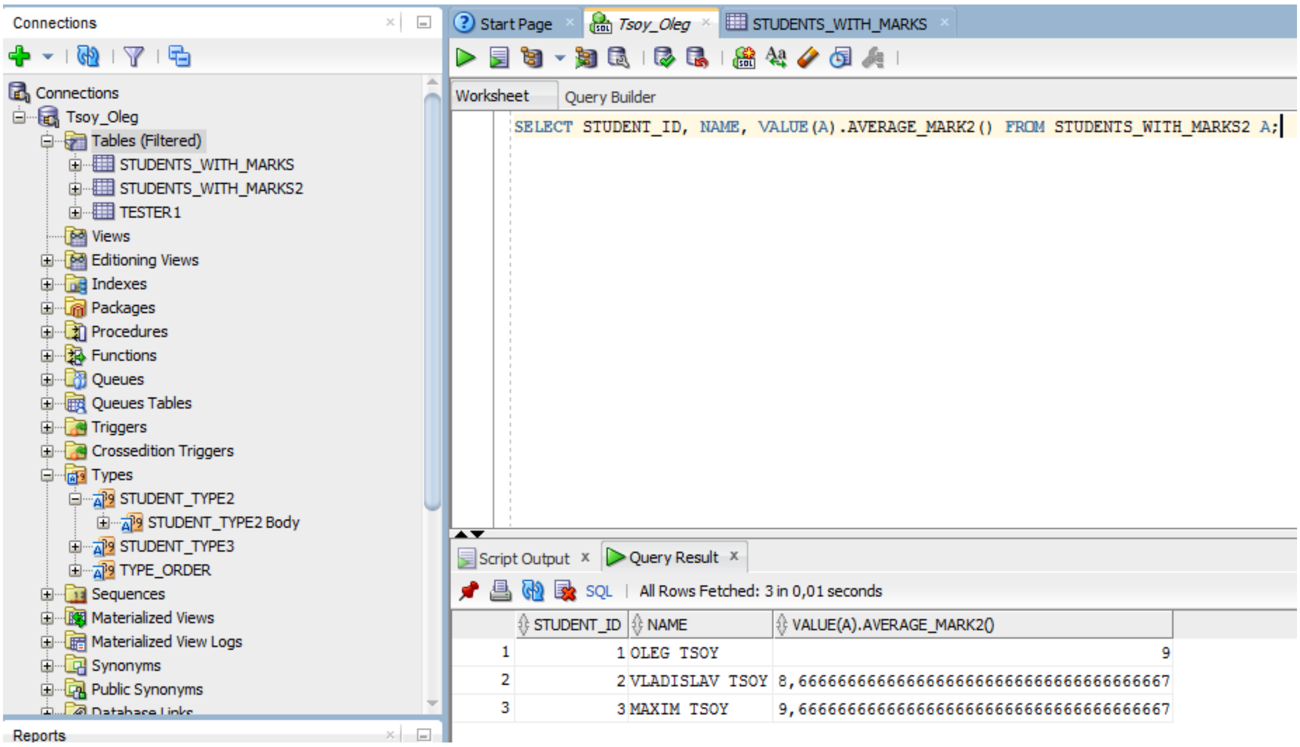
```
1 rows inserted.
1 rows inserted.
1 rows inserted.
```

Analysis of results, what in these data can be seen: *In this screenshot the inserted rows values were indicated for future manipulations.*

13. Query goal (MEMBER type method):
Query SQL code

SELECT STUDENT_ID, NAME, VALUE(A).AVERAGE_MARK2() FROM STUDENTS_WITH_MARKS2 A;

Result of execution



```
SELECT STUDENT_ID, NAME, VALUE(A).AVERAGE_MARK2() FROM STUDENTS_WITH_MARKS2 A;
```

All Rows Fetched: 3 in 0,01 seconds

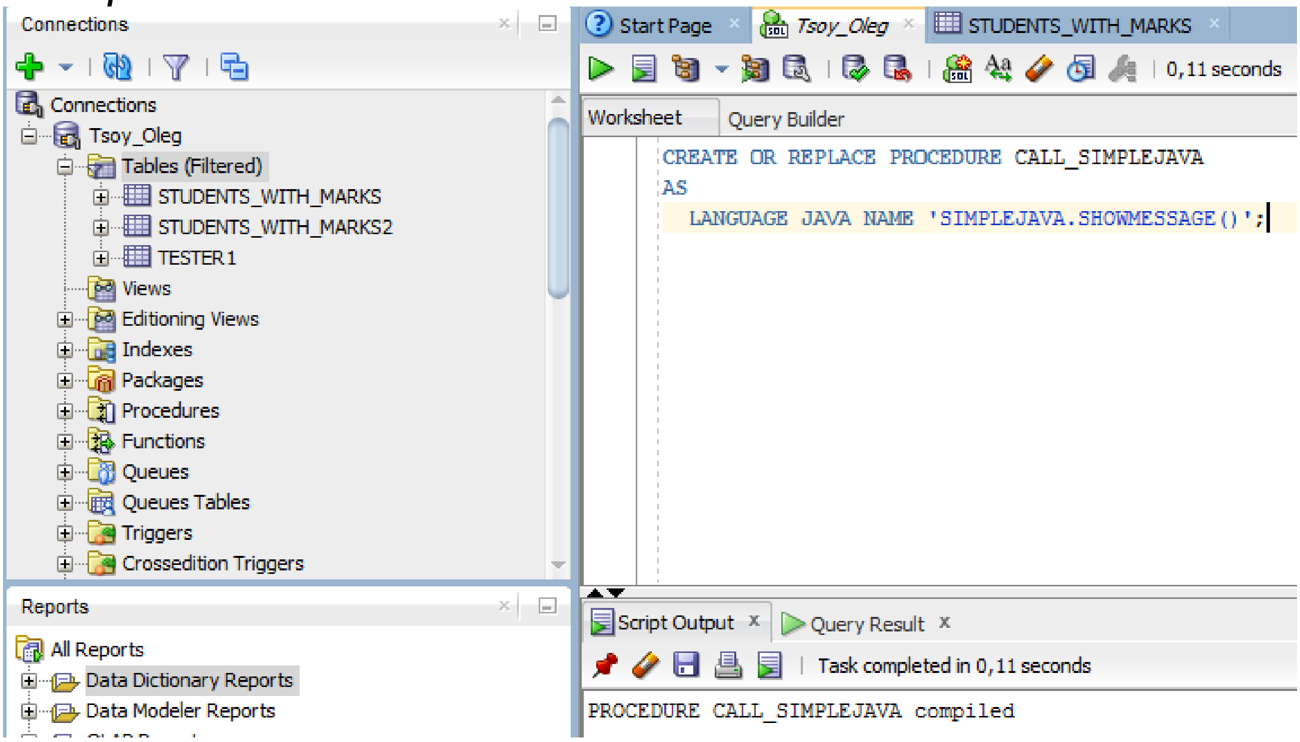
	STUDENT_ID	NAME	VALUE(A).AVERAGE_MARK2()
1	1	OLEG TSOY	9
2	2	VLADISLAV TSOY	8,6666666666666666666666666666666666667
3	3	MAXIM TSOY	9,6666666666666666666666666666666666667

Analysis of results, what in these data can be seen: *The demonstration of inserted data and the performance of the member function working calculations with inserted values.*

14. Query goal (Using of LOADJAVA program):
Query SQL code

```
CREATE OR REPLACE PROCEDURE CALL_SIMPLEJAVA
AS
  LANGUAGE JAVA NAME 'SIMPLEJAVA.SHOWMESSAGE()';
```

Result of execution



```
CREATE OR REPLACE PROCEDURE CALL_SIMPLEJAVA
AS
  LANGUAGE JAVA NAME 'SIMPLEJAVA.SHOWMESSAGE()';
```

Task completed in 0,11 seconds

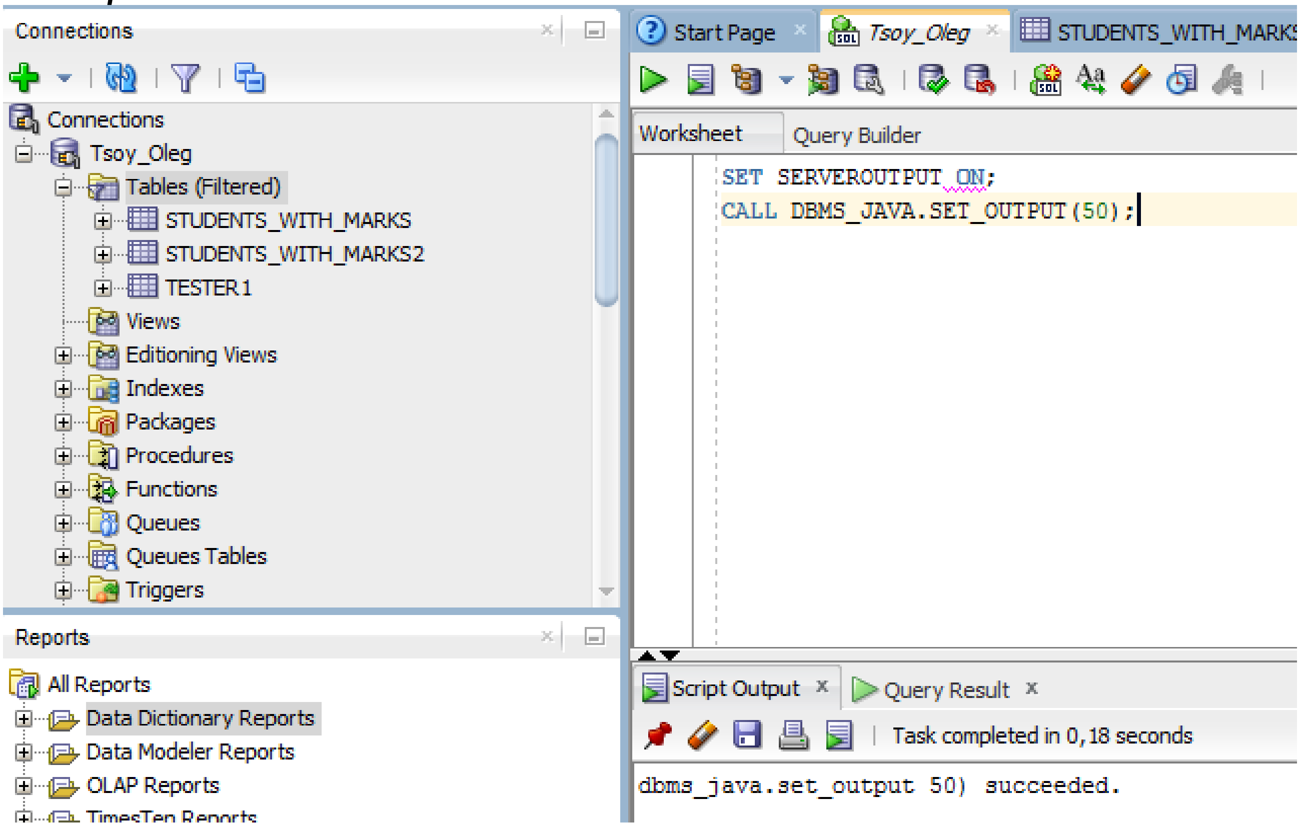
```
PROCEDURE CALL_SIMPLEJAVA compiled
```

Analysis of results, what in these data can be seen: *I have created call_simplejava as language java in my Database system.*

15. Query goal (Using of LOADJAVA program):
Query SQL code

```
SET SERVEROUTPUT ON;
CALL DBMS_JAVA.SET_OUTPUT(50);
```

Result of execution



```
SET SERVEROUTPUT ON;
CALL DBMS_JAVA.SET_OUTPUT(50);
```

Task completed in 0,18 seconds

```
dbms_java.set_output 50) succeeded.
```

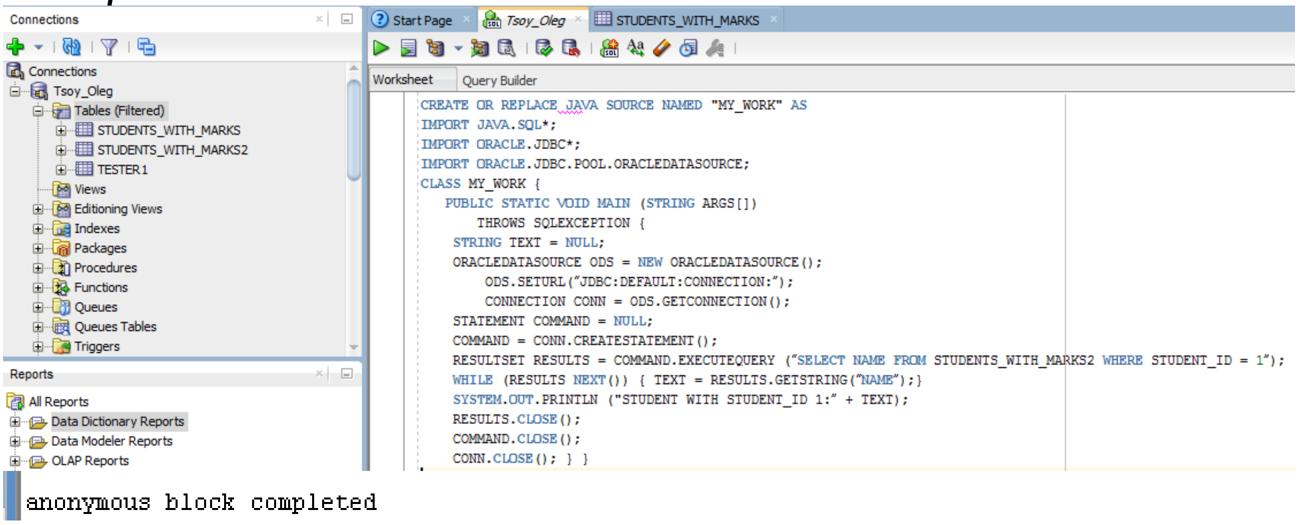
Analysis of results, what in these data can be seen: *The server output by java programming language was succeeded.*

16. Query goal (JAVA SELECT):
Query SQL code

```
CREATE OR REPLACE JAVA SOURCE NAMED "MY_WORK" AS
IMPORT JAVA.SQL*;
IMPORT ORACLE.JDBC*;
IMPORT ORACLE.JDBC.POOL.ORACLEDATASOURCE;
CLASS MY_WORK {
  PUBLIC STATIC VOID MAIN (STRING ARGS[])
     THROWS SQLEXCEPTION {
                 STRING TEXT = NULL;
                 ORACLEDATASOURCE ODS = NEW ORACLEDATASOURCE();
      ODS.SETURL(“JDBC:DEFAULT:CONNECTION:”);
      CONNECTION CONN = ODS.GETCONNECTION();
                 STATEMENT COMMAND = NULL;
                 COMMAND = CONN.CREATESTATEMENT();
```

```
            RESULTSET RESULTS = COMMAND.EXECUTEQUERY (“SELECT NAME
FROM STUDENTS_WITH_MARKS2 WHERE STUDENT_ID = 1”);
            WHILE (RESULTS NEXT()) { TEXT = RESULTS.GETSTRING(“NAME”);}
            SYSTEM.OUT.PRINTLN ("STUDENT WITH STUDENT_ID 1:” + TEXT);
            RESULTS.CLOSE();
            COMMAND.CLOSE();
            CONN.CLOSE(); } }
```

Result of execution



```
CREATE OR REPLACE JAVA SOURCE NAMED "MY_WORK" AS
IMPORT JAVA.SQL*;
IMPORT ORACLE.JDBC*;
IMPORT ORACLE.JDBC.POOL.ORACLEDATASOURCE;
CLASS MY_WORK {
   PUBLIC STATIC VOID MAIN (STRING ARGS[])
       THROWS SQLEXCEPTION {
    STRING TEXT = NULL;
    ORACLEDATASOURCE ODS = NEW ORACLEDATASOURCE();
        ODS.SETURL(“JDBC:DEFAULT:CONNECTION:”);
        CONNECTION CONN = ODS.GETCONNECTION();
    STATEMENT COMMAND = NULL;
    COMMAND = CONN.CREATESTATEMENT();
    RESULTSET RESULTS = COMMAND.EXECUTEQUERY (“SELECT NAME FROM STUDENTS_WITH_MARKS2 WHERE STUDENT_ID = 1”);
    WHILE (RESULTS NEXT()) { TEXT = RESULTS.GETSTRING(“NAME”);}
    SYSTEM.OUT.PRINTLN ("STUDENT WITH STUDENT_ID 1:” + TEXT);
    RESULTS.CLOSE();
    COMMAND.CLOSE();
    CONN.CLOSE(); } }
```

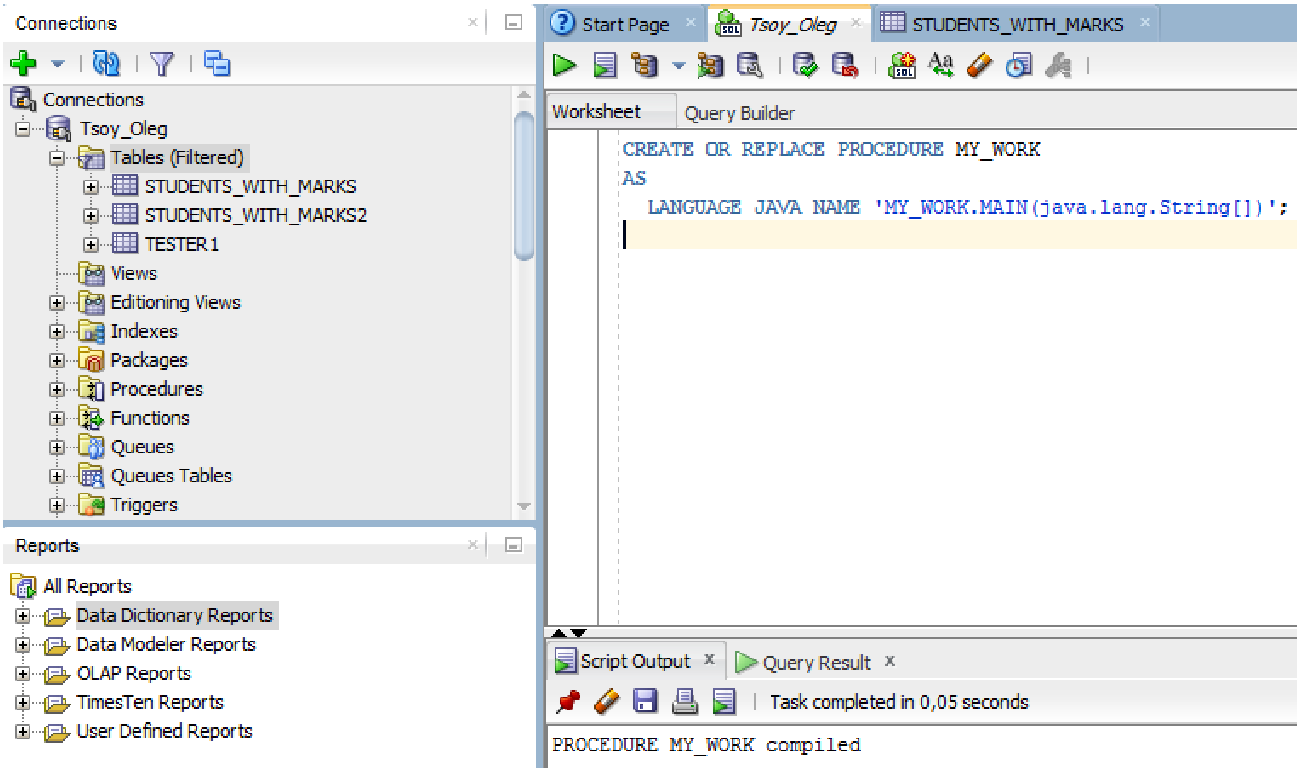
```
anonymous block completed
```

Analysis of results, what in these data can be seen: *Execution of JAVA language meta-data inspection.*

17. Query goal (CREATE JAVA procedure):
Query SQL code

```
CREATE OR REPLACE PROCEDURE MY_WORK
AS
  LANGUAGE JAVA NAME 'MY_WORK.MAIN(java.lang.String[])';
```

Result of execution



```
CREATE OR REPLACE PROCEDURE MY_WORK
AS
  LANGUAGE JAVA NAME 'MY_WORK.MAIN(java.lang.String[])';
```

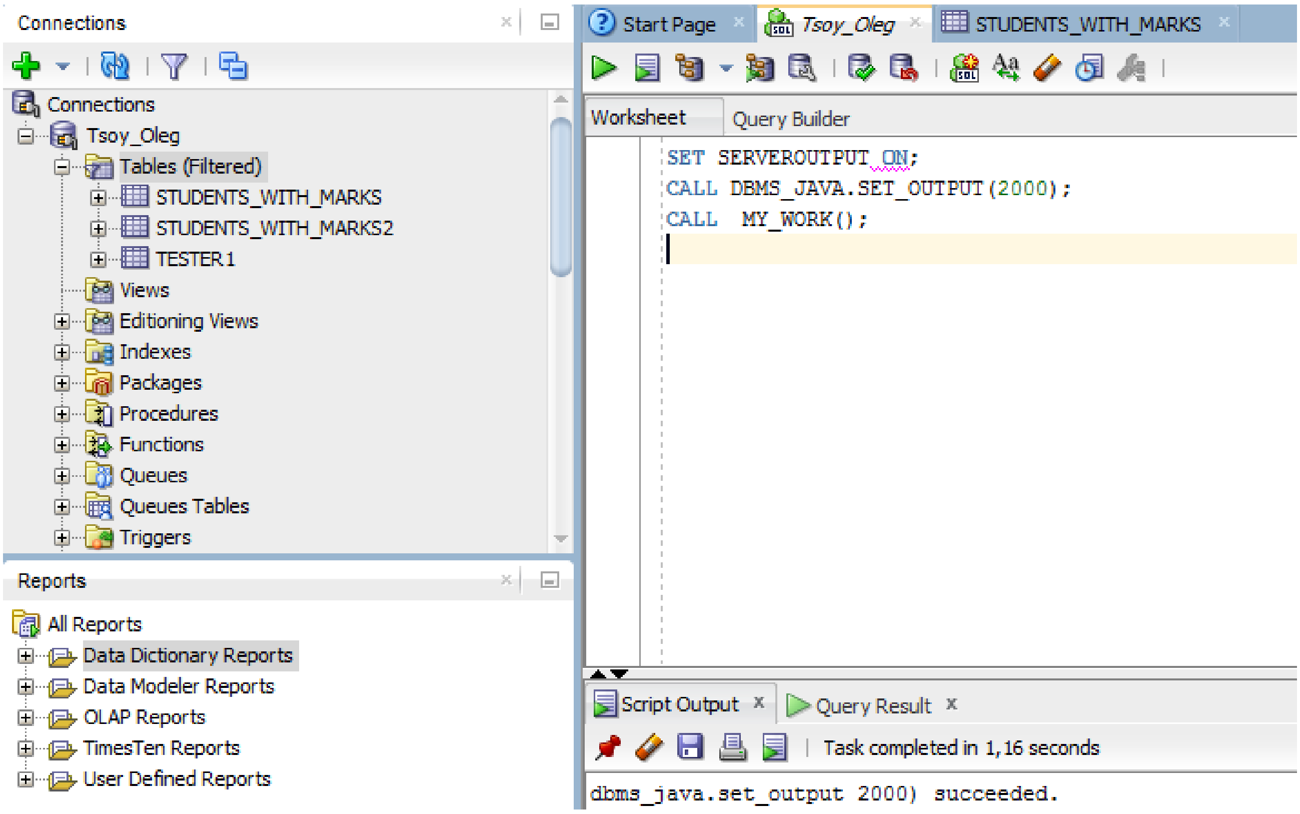
```
PROCEDURE MY_WORK compiled
```

Analysis of results, what in these data can be seen: *Procedure MY_WORK was compiled successfully. That shows that creation of the procedure MY_WORK which is MAIN format has language JAVA as a priority created language.*

18. Query goal (serveroutput of JAVA procedure):
Query SQL code

```
SET SERVEROUTPUT ON;
CALL DBMS_JAVA.SET_OUTPUT(2000);
CALL  MY_WORK();
```

Result of execution



```
SET SERVEROUTPUT ON;
CALL DBMS_JAVA.SET_OUTPUT(2000);
CALL  MY_WORK();
```

Task completed in 1,16 seconds

```
dbms_java.set_output 2000) succeeded.
```

Analysis of results, what in these data can be seen: *In this case we can observe that this query call the execution of MY_WORK procedure.*

4 Conclusion

As can be concluded from this project description I can say that it was the most difficult database project for me because of the existence of JAVA programming language for comparing all data. While, I found some advantages in using of the programming language or IDE which can provide the highest performance in my procedures and simplify the initial code structure. In my case I used Eclipse IDE for compiling JAVA with Oracle Database Management System as can be seen from the most cases in my preview. The problematic part of this host was to find the path how to realize the code description due to the SQL developer and SQL Plus developer have the different structure. However, I was late after reading the materials about them that is why it was my main problem in this case.*