|  |
| --- |
| EPAM Systems, RD Dep. |
| Report 08. Transportation |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| REVISION HISTORY | | | | | |
| Ver. | Description of Change | Author | Date | Approved | |
| Name | Effective Date |
| 1.0 | Initial status | [Alina](mailto:Kiryl_Bucha@epam.com) Makarets | 03-DEC-2017 |  |  |

Contents

[1. Табличные функции 3](#_Toc500186337)

[2. Результаты 4](#_Toc500186338)

# Табличные функции

Табличные функции используются для возврата PL/SQL-коллекций, которые имитируют таблицы. Они могут быть запрошены как обычные таблицы с помощью функцию TABLE во фразе FROM. Обычные табличные функции требуют, чтобы коллекции перед возвращением были полностью наполнены (населены). Так как коллекции хранятся в памяти, это может стать проблемой, поскольку на большие коллекции впустую тратится много памяти и времени в ожидании возвращения первой строки. Обычные табличные функции требуют создания именованной строки и табличных типов как объектов базы данных.

Для начала создадим объект:

--Creating object:

CREATE OR REPLACE TYPE TypeForFunction

AS

OBJECT

(EmpCd NUMBER,

EmpName VARCHAR(60),

DepartCode NUMBER,

Salary NUMBER );/

CREATE TYPE t\_tf\_tab IS TABLE OF t\_tf\_row;

/

CREATE OR REPLACE TYPE TableForFunction AS TABLE OF TypeForFunction;

/

Создадим функцию:

CREATE OR REPLACE

FUNCTION MyFunction

(deptCode NUMBER) RETURN TableForFunction

IS

OutTableParam TableForFunction;

BEGIN

SELECT CAST(

MULTISET(SELECT empno,

ename,

deptno,

sal

FROM scott.emp

WHERE deptno = deptcode)

as TableForFunction)

INTO outtableparam

FROM dual;

RETURN OutTableParam;

END;

/

Проверка

SELECT \* FROM TABLE(MyFunction(10));

# Результаты