

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Емельянов Сергей Геннадьевич

Должность: ректор

Дата подписания: 27.04.2023 09:18:23

Уникальный программный ключ:

9ba7d3e34c012eba476ffd2d064cf2781953be730df2374d16f3c0ce536f0fc6

**МИНОБРАЗОВАНИЯ РОССИИ**

Федеральное государственное бюджетное образовательное  
учреждение высшего образования  
«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра программной инженерии

УТВЕРЖДАЮ

Проректор по учебной работе

О.Г. Локтионова

2020 г.



## ИСПОЛЬЗОВАНИЕ ПЛОЩАДНЫХ ТИПОВ ОБЪЕКТОВ

Методические указания по выполнению лабораторной работы  
по дисциплине «Пространственные базы данных»  
для студентов направления подготовки 09.04.04 «Программная  
инженерия»

Курск 2020

УДК 004.65

Составители: В.Г. Белов, Т.М. Белова

Рецензент

Кандидат технических наук, доцент кафедры программной инженерии ЮЗГУ И.Н. Ефремова

**Использование площадных типов объектов:** методические указания по выполнению лабораторной работы по дисциплине «Пространственные базы данных» для студентов направления подготовки 09.04.04 «Программная инженерия» / Юго-Зап. гос. ун-т; сост.: В.Г. Белов, Т.М. Белова, – Курск, 2020. – 29 с.: ил. 44.

Изложена последовательность действий с Oracle Spatial по использованию площадных типов.

Материал предназначен для студентов направления подготовки 09.04.04 «Программная инженерия», а также будет полезен студентам всех направлений подготовки, изучающим технологии разработки пространственных баз данных.

Текст печатается в авторской редакции.

Подписано в печать 18.04.20. Формат 60x84 1/16.

Усл. печ. л. 1,7. Уч.-изд. л. 1,5. Тираж 100 экз. Заказ 4443. Бесплатно.

Юго-Западный государственный университет.

305040, Курск, ул.50 лет Октября, 94.

## Содержание

1 Цель лабораторной работы .....	4
2 Порядок выполнения лабораторной работы .....	5
3 Содержание отчета по лабораторной работе .....	28
4 Вопросы к защите лабораторной работы .....	29

## **1 Цель лабораторной работы**

Целью лабораторной работы является приобретение знаний умений и навыков при задании параметров площадных объектов.

Данные действия осуществляет разработчик базы данных, который проектирует структуру будущей базы данных. Задание параметров площадных объектов является важным навыком при работе с подсистемой СУБД Oracle Spatial.

## 2 Порядок выполнения лабораторной работы

Для задания параметров площадных объектов необходимо создать таблицу, в которой будет находиться информация об этих объектах. Назовем данную таблицу COLA\_MARKETS. Эта таблица будет хранить информацию о фирменных магазинах фирмы Coca-Cola.

На рисунке 1 представлены SQL выражения для создания таблицы COLA\_MARKETS, которые добавляют мета-данные в представление USER\_SDO\_GEOMETRY и создают пространственный индекс для столбца SDO\_GEOMETRY. Данные выражения можно выполнить в среде JDeveloper или в утилите SQL\*Plus.

```

CREATE TABLE cola_markets (
mkt_id NUMBER PRIMARY KEY,
CREATE TABLE cola_markets (
    mkt_id NUMBER PRIMARY KEY,
    name VARCHAR2(32),
    shape SDO_GEOMETRY
);

INSERT INTO user_sdo_geom_metadata (
    TABLE_NAME,
    COLUMN_NAME,
    DIMINFO,
    SRID
)
VALUES (
    'cola_markets',
    'shape',
    SDO_DIM_ARRAY( -- 20X20 grid
        SDO_DIM_ELEMENT('X', 0, 20, 0.005),
        SDO_DIM_ELEMENT('Y', 0, 20, 0.005)
    ),
    NULL -- SRID
);

CREATE INDEX cola_spatial_idx ON cola_markets(shape)
INDEXTYPE IS MDSYS.SPATIAL_INDEX;

```

Рис. 1 – SQL выражения для создания таблицы COLA\_MARKETS

Зададим параметры полигона, изображенного на рисунке 2.

Для этого введите SQL выражения, представленные на рисунке 3.

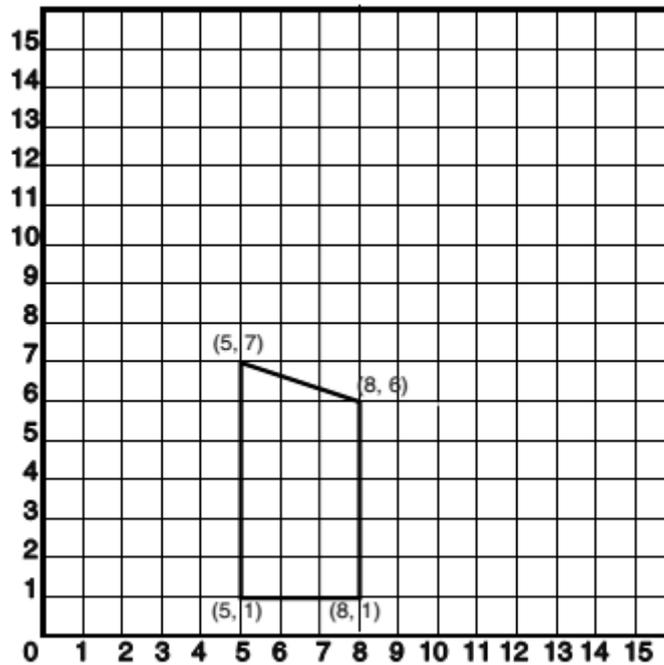


Рис. 2 – Полигон

```

INSERT INTO cola_markets VALUES(
301,
'polygon',
SDO_GEOMETRY(
2003, -- two-dimensional polygon
NULL,
NULL,
SDO_ELEM_INFO_ARRAY(1,1003,1), -- one polygon (exterior polygon ring)
SDO_ORDINATE_ARRAY(5,1, 8,1, 8,6, 5,7, 5,1)
)
);

```

Рис. 3 – SQL выражения для задания параметров полигона,  
изображенного на рисунке 2

В SQL выражениях, которые изображена на рисунке 3, задаются следующие параметры полигона:

- SDO\_GTYPE=2003. 2 индицирует 2 измерения, а 3 индицирует полигон.
- SDO\_GRID=NULL.

- `SDO_POINT=NULL`.
- `SDO_ELEM_INFO=(1, 1003, 1)`. 1 в 1003 индицирует внешний контур полигона. Последняя 1 в 1, 1003, 1 индицирует, что это простой полигон, вершины которого напрямую соединены отрезками и то, что вы должны определить координаты для каждой точки вершины, координаты последней вершины должны быть такими же как и первой.
- `SDO_COORDINATES=(5,1, 8,1, 8,6, 5,7, 5,1)`. Это идентифицирует вершины полигона с одинаковыми первыми и последними координатами. Потому что это полигон с внешним контуром (простой полигон не имеет внутреннего контура) координаты расположены по часовой стрелке.

Зададим параметры прямоугольника, изображенного на рисунке 4. Для этого введите SQL выражения, представленные на рисунке 5.

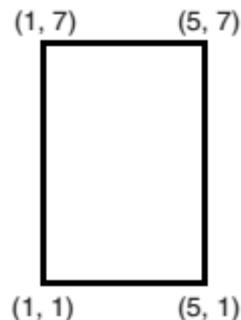


Рис. 4 – Прямоугольник

```

INSERT INTO cola_markets VALUES(
302,
'rectangle',
SDO_GEOMETRY(
2003, -- two-dimensional polygon
NULL,
NULL,
SDO_ELEM_INFO_ARRAY(1,1003,3), -- one rectangle (1003 = exterior)
SDO_ORDINATE_ARRAY(1,1, 5,7) -- only 2 points needed to
-- define rectangle (lower left and upper right) with
-- Cartesian-coordinate data
)
);

```

Рис. 5 - SQL выражения для задания параметров  
прямоугольника, изображенного на рисунке 4

В SQL выражениях, которые изображена на рисунке 5, задаются следующие параметры прямоугольника:

- SDO\_GTYPE=2003. 2 индицирует 2 измерения, а 3 индицирует полигон.
- SDO\_GRID=NULL.
- SDO\_POINT=NULL.
- SDO\_ELEM\_INFO=(1, 1003, 3). Последняя 3 в 1, 1003, 3 индицирует, что это прямоугольник. Так как это прямоугольник, только две координаты указаны в SDO\_ORDINATES (левая нижняя и верхняя правая).
- SDO\_ORDINATES = (1,1, 5,7). Это идентифицирует левую нижнюю и правую верхнюю координаты.

Зададим параметры полигона с отверстием, изображенного на рисунке 6. Для этого введите SQL выражения, представленные на рисунке 7.

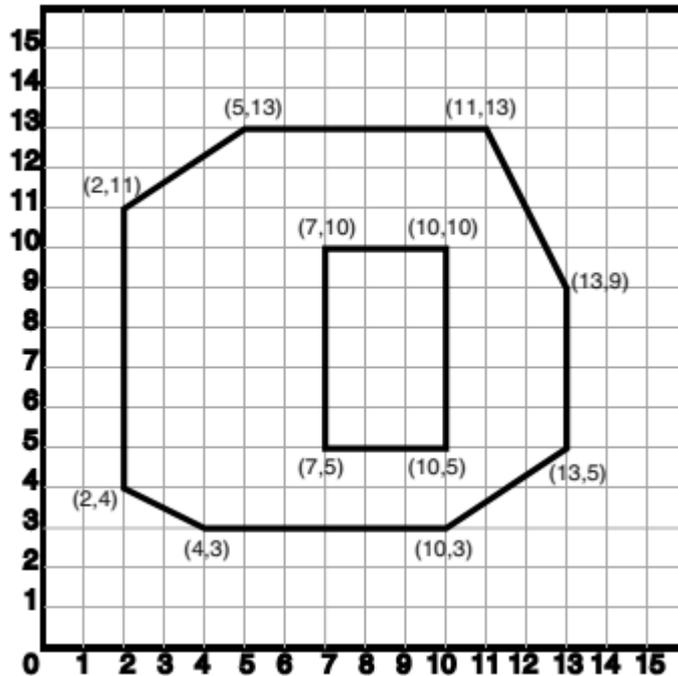


Рис. 6 – Полигон с отверстием

```
INSERT INTO cola_markets VALUES(
303,
'polygon_with_hole',
SDO_GEOMETRY(
2003, -- two-dimensional polygon
NULL,
NULL,
SDO_ELEM_INFO_ARRAY(1,1003,1, 19,2003,1), -- polygon with hole
SDO_ORDINATE_ARRAY(2,4, 4,3, 10,3, 13,5, 13,9, 11,13, 5,13, 2,11, 2,4,
7,5, 7,10, 10,10, 10,5, 7,5)
)
);
```

Рис. 7 - SQL выражения для задания параметров полигона с отверстием, изображенного на рисунке 6

В SQL выражениях, которые изображена на рисунке 7, задаются следующие параметры многоугольника:

- SDO\_GTYPE=2003. 2 индицирует 2 измерения, а 3 индицирует полигон.
- SDO\_GRID=NULL.
- SDO\_POINT=NULL.
- SDO\_ELEM\_INFO=(1,1003,1, 19,2003,1). Здесь находятся два триплета 1,1003,1 и 19,2003,1.
  - 1003 индицирует, что это внешний полигон;
  - 2003 индицирует, что это внутренний полигон;
  - 19 индицирует, что координаты второго (внутреннего) полигона начинаются с 19 позиции в массиве SDO\_ORDINATES (это 7, что означает, что первая координата равна 7,5).
- SDO\_ORDINATES = (2,4, 4,3, 10,3, 13,5, 13,9, 11,13, 5,13, 2,11, 2,4, 7,5, 7,10, 10,10, 10,5, 7,5).
- Площадь этого прямоугольника (функция SDO\_GEOM.AREA) равна разнице площадей внутреннего и внешнего полигонов.
- Периметр этого полигона (функция SDO\_GEOM.LENGTH) равен сумме периметров внутреннего и внешнего полигонов.

Например, такой полигон может быть участком земли, такой как страна или остров, внутри которого находится озеро. Конечно

реальный участок земли может иметь много таких внутренних полигонов: каждый должен быть представлен триплетом в SDO\_ELEM\_INFO и еще необходимыми координатами.

Внутренний и внешний полигоны не могут быть вложенными. Например, если в стране есть озеро и в нем есть остров, и в этом острове есть озеро. В этом случае отдельный полигон должен быть определен для острова, так как остров с озером не может быть определен как внутренний полигон, внутри которого определен еще один полигон, представляющей собой озеро.

В наборе полигонов контуры должны быть сгруппированы полигоном, и первый контур в каждом полигоне должен быть внешним полигоном. Для примера рассмотрим набор полигонов, который содержит 2 полигона А и В:

- Полигон А (одна внутренняя «дыра»): внешний полигон – А0, а внутренний – А1.
- Полигон В (две внутренних «дыры»): внешний полигон – В0, внутренний полигон – В1 и внутренний полигон – В2.

Элементы в SDO\_ELEM\_INFO и в SDO\_ORDINATES должны быть в одном из следующих порядков (зависит от того какой полигон вы захотите указать первым – А или В):

- А0, А1; В0, В1, В2.
- В0, В1, В2; А0, А1.

Зададим параметры ломаной линии, изображенной на рисунке 8. Для этого введите SQL выражения, представленные на рисунке 9. Для описания данной фигуры необходимы 4 точки: (3,2), (4,6), (6,4), и (14,7).

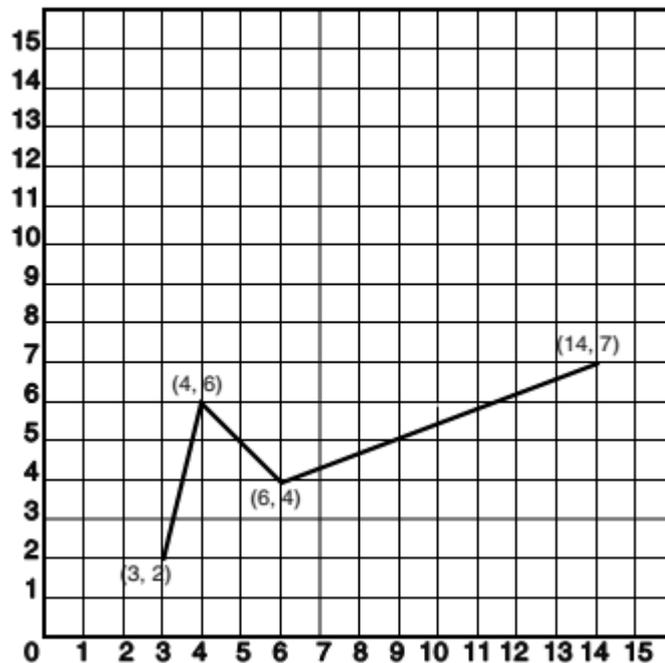


Рис. 8 – Ломаная линия

```
INSERT INTO cola_markets VALUES(  
304,  
'line_string',  
SDO_GEOMETRY(  
2002,  
NULL,  
NULL,  
SDO_ELEM_INFO_ARRAY(1,2,1), -- line string  
SDO_ORDINATE_ARRAY(3,2, 4,6, 6,4, 14,7)  
)  
);
```

Рис. 9 - SQL выражения для задания параметров ломаной линии, изображенной на рисунке 8

В SQL выражениях, которые изображена на рисунке 9, задаются следующие параметры ломаной линии:

- SDO\_GTYPE=2002. Первая 2 индицирует 2 измерения, а вторая индицирует один и более отрезков.
- SDO\_GRID=NULL.
- SDO\_POINT=NULL.
- SDO\_ELEM\_INFO=(1,2,1). 2,1 в 1,2,1 индицирует ломаную линию, чьи вершины соединяются отрезками.
- SDO\_ORDINATES= (3,2, 4,6, 6,4, 14,7).

Зададим параметры лунообразного объекта, изображенного на рисунке 10. Данный объект является составной ломаной линией, которая создана из отрезка и круговой дуги. Для этого введите SQL выражения, представленные на рисунке 11. Необходимо 4 точки чтобы описать эту фигуру: точки (10,10) и (10,14) описывают отрезок, а точки (10,14), (6,10) и (14,10) описывают круговую дугу.

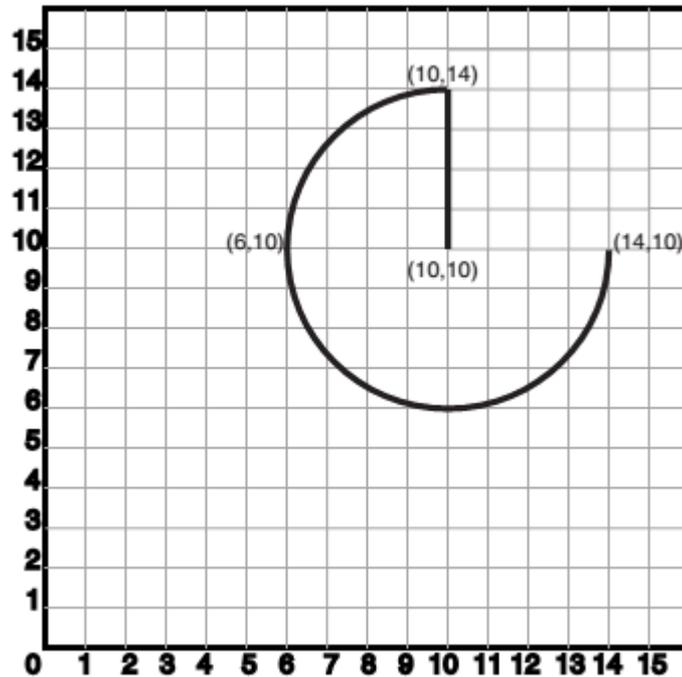


Рис. 10 – Составная ломаная линия

```

INSERT INTO cola_markets VALUES(
305,
'compound_line_string',
SDO_GEOMETRY(
2002,
NULL,
NULL,
SDO_ELEM_INFO_ARRAY(1,4,2, 1,2,1, 3,2,2), -- compound line string
SDO_ORDINATE_ARRAY(10,10, 10,14, 6,10, 14,10)
)
);

```

Рис. 11 - SQL выражения для задания параметров составной ломаной линии, изображенной на рисунке 10

В SQL выражениях, которые изображены на рисунке 11, задаются следующие параметры составной ломаной линии:

- SDO\_GTYPE=2002. Первая 2 индицирует 2 измерения, а вторая индицирует один и более отрезков.
- SDO\_GRID=NULL.

- `SDO_POINT=NULL`.
- `SDO_ELEM_INFO = (1,4,2, 1,2,1, 3,2,2)`. Здесь три триплета:
  - 1,4,2. Этот триплет индицирует, что эта фигура является составной, и состоит она из двух подэлементов, которые будут объяснены в следующих двух триплетах.
  - 1,2,1. Этот триплет индицирует, что линия состоит из двух сегментов, координаты которых начинаются с первого элемента `SDO_ORDINATES`. Последняя точка для этого отрезка определяется началом смещения второго отрезка, в данном случае – 3.
  - 3,2,2. Этот триплет индицирует, что вторая линия представляет собой круговую дугу, координаты которой начинаются с позиции 3. Последняя точка этого отрезка определена началом смещения следующего элемента или длиной массива `SDO_ORDINATES`, если это последний элемент.
- `SDO_ORDINATES = (10,10, 10,14, 6,10, 14,10)`.

#### Составной полигон

Зададим параметры составного полигона, изображенного на рисунке 12. Для этого введите SQL выражения, представленные на рисунке 13.

Данная фигура представляет собой конусообразный объект, представленный как составной полигон, состоящий из одной ломанной линии и одной круговой дуги. Необходимо пять точек, чтобы представить этот объект: точки (6,10), (10, 1) и (14, 10) описывают остроугольную ломанную линию, а точки (14, 10), (10, 14) и (6, 10) описывают круговую дугу. Начальная точка ломанной линии и последняя точка дуги совпадают (координата (6, 10)). SDO\_ELEM\_INFO состоит из трех триплетов: (1,1005,2), (1,2,1) и (5,2,2).

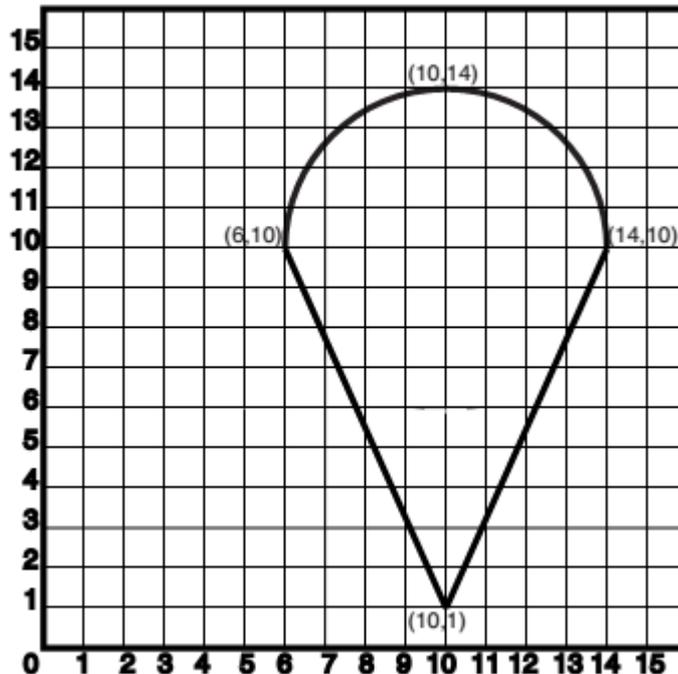


Рис. 12 – Составной полигон

```

INSERT INTO cola_markets VALUES(
306,
'compound_polygon',
SDO_GEOMETRY(
2003, -- two-dimensional polygon
NULL,
NULL,
SDO_ELEM_INFO_ARRAY(1,1005,2, 1,2,1, 5,2,2), -- compound polygon
SDO_ORDINATE_ARRAY(6,10, 10,1, 14,10, 10,14, 6,10)
)
);

```

Рис. 13 - SQL выражения для задания параметров составного полигона, изображенного на рисунке 12

В SQL выражениях, которые изображены на рисунке 13, задаются следующие параметры составного полигона:

- SDO\_GTYPE=2003. 2 индицирует 2 измерения, а 3 индицирует полигон.
- SDO\_GRID=NULL.
- SDO\_POINT=NULL.
- SDO\_ELEM\_INFO = (1,1005,2, 1,2,1, 5,2,2). Здесь содержится 3 триплета:
  - 1,1005,2. Этот триплет индицирует, что эта фигура, состоящая из подэлементов, который описаны в следующих двух триплетах.
  - 1,2,1. Этот триплет индицирует, что первый подэлемент представляет собой ломанную линию, координаты которой начинаются с первого элемента в массиве SDO\_ORDINATES.

- 5,2,2. Этот триплет индицирует, что второй полэлемент представляет собой круговую дугу, координаты которой начинаются с пятого элемента в массиве SDO\_ORDINATES. Последняя точка этой фигуры описана как начальное смещение следующего элемента или текущий размер массива SDO\_ORDINATES.
  - SDO\_ORDINATES = (6,10, 10,1, 14,10, 10,14, 6,10).

Примеры на рисунках 14 – 44 создают таблицу и вставляют в неё различные объекты, включая кластеры точек, наборы полигонов и коллекции фигур. В конце вызывается функция SDO\_GEOM.VALIDATE\_GEOMETRY\_WITH\_CONTEXT, чтобы проверить вставленные объекты. Обратите внимание, что некоторые объекты умышленно определены неверно. В именах таких объектов включено слово INVALID.

```
CREATE TABLE t1 (  
  i NUMBER,  
  d VARCHAR2(50),  
  g SDO_GEOMETRY  
);  
INSERT INTO t1 (i, d, g)  
VALUES (  
  1,  
  'Line segment',  
  sdo_geometry (2002, null, null, sdo_elem_info_array (1,2,1),  
  sdo_ordinate_array (10,10, 20,10))  
);
```

Рис. 14

```
INSERT INTO t1 (i, d, g)  
VALUES (  
  2,  
  'Arc segment',  
  sdo_geometry (2002, null, null, sdo_elem_info_array (1,2,2),  
  sdo_ordinate_array (10,15, 15,20, 20,15))  
);  
INSERT INTO t1 (i, d, g)  
VALUES (  
  3,  
  'Line string',  
  sdo_geometry (2002, null, null, sdo_elem_info_array (1,2,1),  
  sdo_ordinate_array (10,25, 20,30, 25,25, 30,30))  
);
```

Рис. 15

```
INSERT INTO t1 (i, d, g)  
VALUES (  
  4,  
  'Arc string',  
  sdo_geometry (2002, null, null, sdo_elem_info_array (1,2,2),  
  sdo_ordinate_array (10,35, 15,40, 20,35, 25,30, 30,35))  
);
```

Рис. 16

```
INSERT INTO t1 (i, d, g)
VALUES (
5,
'Compound line string',
sdo_geometry (2002, null, null,
sdo_elem_info_array (1,4,3, 1,2,1, 3,2,2, 7,2,1),
sdo_ordinate_array (10,45, 20,45, 23,48, 20,51, 10,51))
);
```

Рис. 17

```
INSERT INTO t1 (i, d, g)
VALUES (
6,
'Closed line string',
sdo_geometry (2002, null, null, sdo_elem_info_array (1,2,1),
sdo_ordinate_array (10,55, 15,55, 20,60, 10,60, 10,55))
);
```

Рис. 18

```
INSERT INTO t1 (i, d, g)
VALUES (
7,
'Closed arc string',
sdo_geometry (2002, null, null, sdo_elem_info_array (1,2,2),
sdo_ordinate_array (15,65, 10,68, 15,70, 20,68, 15,65))
);
```

Рис. 19

```
INSERT INTO t1 (i, d, g)
VALUES (
8,
'Closed mixed line',
sdo_geometry (2002, null, null, sdo_elem_info_array (1,4,2, 1,2,1, 7,2,2),
sdo_ordinate_array (10,78, 10,75, 20,75, 20,78, 15,80, 10,78))
);
```

Рис. 20

```
INSERT INTO t1 (i, d, g)
VALUES (
9,
'Self-crossing line',
sdo_geometry (2002, null, null, sdo_elem_info_array (1,2,1),
sdo_ordinate_array (10,85, 20,90, 20,85, 10,90, 10,85))
);
```

Рис. 21

```
INSERT INTO t1 (i, d, g)
VALUES (
10,
'Polygon',
sdo_geometry (2003, null, null, sdo_elem_info_array (1,1003,1),
sdo_ordinate_array (10,105, 15,105, 20,110, 10,110, 10,105))
);
```

Рис. 22

```
INSERT INTO t1 (i, d, g)
VALUES (
11,
'Arc polygon',
sdo_geometry (2003, null, null, sdo_elem_info_array (1,1003,2),
sdo_ordinate_array (15,115, 20,118, 15,120, 10,118, 15,115))
);
```

Рис. 23

```
INSERT INTO t1 (i, d, g)
VALUES (
12,
'Compound polygon',
sdo_geometry (2003, null, null, sdo_elem_info_array (1,1005,2, 1,2,1, 7,2,2),
sdo_ordinate_array (10,128, 10,125, 20,125, 20,128, 15,130, 10,128))
);
```

Рис. 24

```
INSERT INTO t1 (i, d, g)
VALUES (
13,
'Rectangle',
sdo_geometry (2003, null, null, sdo_elem_info_array (1,1003,3),
sdo_ordinate_array (10,135, 20,140))
);
```

Рис. 25

```
INSERT INTO t1 (i, d, g)
VALUES (
14,
'Circle',
sdo_geometry (2003, null, null, sdo_elem_info_array (1,1003,4),
sdo_ordinate_array (15,145, 10,150, 20,150))
);
```

Рис. 26

```
INSERT INTO t1 (i, d, g)
VALUES (
15,
'Point cluster',
sdo_geometry (2005, null, null, sdo_elem_info_array (1,1,3),
sdo_ordinate_array (50,5, 55,7, 60,5))
);
```

Рис. 27

```
INSERT INTO t1 (i, d, g)
VALUES (
16,
'Multipoint',
sdo_geometry (2005, null, null, sdo_elem_info_array (1,1,1, 3,1,1, 5,1,1),
sdo_ordinate_array (65,5, 70,7, 75,5))
);
```

Рис. 28

```
INSERT INTO t1 (i, d, g)
VALUES (
17,
'Multiline',
sdo_geometry (2006, null, null, sdo_elem_info_array (1,2,1, 5,2,1),
sdo_ordinate_array (50,15, 55,15, 60,15, 65,15))
);
```

Рис. 29

```
INSERT INTO t1 (i, d, g)
VALUES (
18,
'Multiline - crossing',
sdo_geometry (2006, null, null, sdo_elem_info_array (1,2,1, 5,2,1),
sdo_ordinate_array (50,22, 60,22, 55,20, 55,25))
);
```

Рис. 30

```
INSERT INTO t1 (i, d, g)
VALUES (
19,
'Multiarc',
sdo_geometry (2006, null, null, sdo_elem_info_array (1,2,2, 7,2,2),
sdo_ordinate_array (50,35, 55,40, 60,35, 65,35, 70,30, 75,35))
);
```

Рис. 31

```
INSERT INTO t1 (i, d, g)
VALUES (
20,
'Multiline - closed',
sdo_geometry (2006, null, null, sdo_elem_info_array (1,2,1, 9,2,1),
sdo_ordinate_array (50,55, 50,60, 55,58, 50,55, 56,58, 60,55, 60,60, 56,58))
);
```

Рис. 32

```
INSERT INTO t1 (i, d, g)
VALUES (
21,
'Multiarc - touching',
sdo_geometry (2006, null, null, sdo_elem_info_array (1,2,2, 7,2,2),
sdo_ordinate_array (50,65, 50,70, 55,68, 55,68, 60,65, 60,70))
);
```

Рис. 33

```
INSERT INTO t1 (i, d, g)
VALUES (
22,
'Multipolygon - disjoint',
sdo_geometry (2007, null, null, sdo_elem_info_array (1,1003,1, 11,1003,3),
sdo_ordinate_array (50,105, 55,105, 60,110, 50,110, 50,105, 62,108, 65,112))
);
```

Рис. 34

```
INSERT INTO t1 (i, d, g)
VALUES (
23,
'Multipolygon - touching',
sdo_geometry (2007, null, null, sdo_elem_info_array (1,1003,3, 5,1003,3),
sdo_ordinate_array (50,115, 55,120, 55,120, 58,122))
);
```

Рис. 35

```
INSERT INTO t1 (i, d, g)
VALUES (
24,
'Multipolygon - tangent * INVALID 13351',
sdo_geometry (2007, null, null, sdo_elem_info_array (1,1003,3, 5,1003,3),
sdo_ordinate_array (50,125, 55,130, 55,128, 60,132))
);
```

Рис. 36

```

INSERT INTO t1 (i, d, g)
VALUES (
25,
'Multipolygon - multi-touch',
sdo_geometry (2007, null, null, sdo_elem_info_array (1,1003,1, 17,1003,1),
sdo_ordinate_array (50,95, 55,95, 53,96, 55,97, 53,98, 55,99, 50,99, 50,95,
55,100, 55,95, 60,95, 60,100, 55,100))
);

```

Рис. 37

```

INSERT INTO t1 (i, d, g)
VALUES (
26,
'Polygon with void',
sdo_geometry (2003, null, null, sdo_elem_info_array (1,1003,3, 5,2003,3),
sdo_ordinate_array (50,135, 60,140, 51,136, 59,139))
);

```

Рис. 38

```

INSERT INTO t1 (i, d, g)
VALUES (
27,
'Polygon with void - reverse',
sdo_geometry (2003, null, null, sdo_elem_info_array (1,2003,3, 5,1003,3),
sdo_ordinate_array (51,146, 59,149, 50,145, 60,150))
);

```

Рис. 39

```

INSERT INTO t1 (i, d, g)
VALUES (
28,
'Crescent (straight lines) * INVALID 13349',
sdo_geometry (2003, null, null, sdo_elem_info_array (1,1003,1),
sdo_ordinate_array (10,175, 10,165, 20,165, 15,170, 25,170, 20,165,
30,165, 30,175, 10,175))
);

```

Рис. 40

```

INSERT INTO t1 (i, d, g)
VALUES (
29,
'Crescent (arcs) * INVALID 13349',
sdo_geometry (2003, null, null, sdo_elem_info_array (1,1003,2),
sdo_ordinate_array (14,180, 10,184, 14,188, 18,184, 14,180, 16,182,
14,184, 12,182, 14,180))
);

```

Рис. 41

```

INSERT INTO t1 (i, d, g)
VALUES (
30,
'Heterogeneous collection',
sdo_geometry (2004, null, null, sdo_elem_info_array (1,1,1, 3,2,1, 7,1003,1),
sdo_ordinate_array (10,5, 10,10, 20,10, 10,105, 15,105, 20,110, 10,110,
10,105))
);

```

Рис. 42

```

INSERT INTO t1 (i, d, g)
VALUES (
31,
'Polygon+void+island touch',
sdo_geometry (2007, null, null,
sdo_elem_info_array (1,1003,1, 11,2003,1, 31,1003,1),
sdo_ordinate_array (50,168, 50,160, 55,160, 55,168, 50,168, 51,167,
54,167, 54,161, 51,161, 51,162, 52,163, 51,164, 51,165, 51,166, 51,167,
52,166, 52,162, 53,162, 53,166, 52,166))
);

```

Рис. 43

```

COMMIT;
SELECT i, d, SDO_GEOM.VALIDATE_GEOMETRY_WITH_CONTEXT (g, 0.5) FROM t1;

```

Рис. 44

### **3 Содержание отчета по лабораторной работе**

В сводный отчет по лабораторным работам в качестве одного из разделов или подразделов включаются скриншоты, показывающие содержимое таблиц при выполнении индивидуального задания.

#### **4 Вопросы к защите лабораторной работы**

1. Как в подсистеме СУБД Oracle Spatial задаются параметры полигона?
2. Как в подсистеме СУБД Oracle Spatial задаются параметры прямоугольника?
3. Как в подсистеме СУБД Oracle Spatial задаются параметры полигона с отверстием?
4. Как в подсистеме СУБД Oracle Spatial задаются параметры ломаной линии?
5. Как в подсистеме СУБД Oracle Spatial задаются параметры составной ломаной линии?
6. Как в подсистеме СУБД Oracle Spatial задаются параметры составного прямоугольника?
7. Могут ли два полигона с отверстиями быть описаны как вложенные?