# Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации

Федеральное государственное образовательное бюджетное учреждение высшего образования «Сибирский государственный университет телекоммуникаций и информатики» (ФГОБУ ВО «СибГУТИ»)
Кафедра ТСиВС

# Практическое занятие №1

Дисциплина: «Технологии проектирования РАСУ»

#### Выполнил:

Студент группы ИА-032

Ющенко А.В.

# Проверил:

Старший преп. каф. ТСиВС

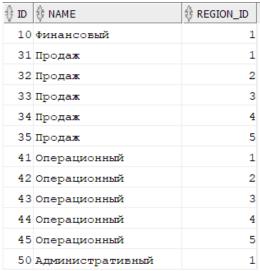
Андреев А.В.

# Выполнение работы

#### Пример 1

Вывести содержимое всех строк и столбцов таблицы DEPT.

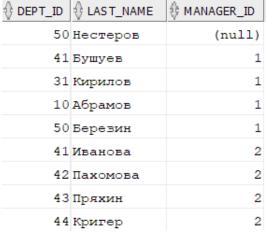
SELECT \* FROM DEPT



#### Пример 2

вывести номера отделов, фамилии служащих и идентификационные номера их менеджеров.

SELECT dept\_id, last\_name, manager\_id FROM EMP



#### Пример 3

вывести заработную плату и комиссионные проценты за год всех сотрудников из таблицы еmp

SELECT salary, commission\_pct FROM EMP

SALARY	⊕ COMMISSION_PCT
5500	(null)
3700	(null)
3500	(null)
3700	(null)
3850	(null)
2200	(null)
2000	(null)
1400	10
1490	12,5
1515	10

вывести фамилию, зарплату и годовые компенсационные выплаты для всех, так чтобы сумма выплат начислялась как зарплата\*12+одноразовая премия=100\$

SELECT last\_name, salary, (salary \* 12 + 100) AS yearly\_compensation FROM EMP

↓ LAST_NAME	\$ SALARY	
Нестеров	5500	66100
Бушуев	3700	44500
Кирилов	3500	42100
Абрамов	3700	44500
Березин	3850	46300
Иванова	2200	26500
Пахомова	2000	24100

#### Пример 5

вывести фамилию, зарплату и сумму выплат для каждого служащего, причем размер выплат вычисляется как прибавление к заработной плате премии 100\$ и умножения этой суммы на 12.

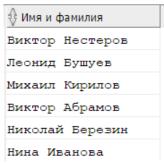
SELECT last\_name, salary, (salary + 100) \* 12 AS total\_payment FROM EMP

↓ LAST_NAME	\$ SALARY	♦ TOTAL_PAYMENT
Нестеров	5500	67200
Бушуев	3700	45600
Кирилов	3500	43200
Абрамов	3700	45600
Березин	3850	47400
Иванова	2200	27600
Пахомова	2000	25200
Пряхин	2000	25200
Кригер	2000	25200
Селедцова	2000	25200

#### Пример 8

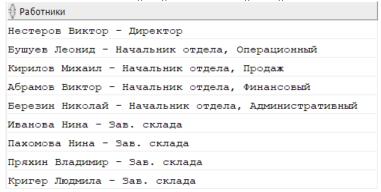
вывод имен и фамилий, служащих под общим заголовком.

SELECT first\_name || ' ' || last\_name AS "Имя и фамилия" FROM EMP



вывести фамилию, имя, должность каждого служащего под заголовком "Работники" с применением знаков пунктуации.

SELECT last\_name || ' ' || first\_name || ' - ' || title AS "Работники" FROM EMP



#### Пример 10

вывести фамилию, зарплату, должность и значение вычисленных комиссионных пропентов.

SELECT last\_name, salary, title, (salary \* commission\_pct / 100) AS calc\_commission FROM EMP

↓ LAST_NAME	SALARY	♦ TITLE	<pre></pre>
Нестеров	5500	Директор	(null)
Бушуев	3700	Начальник отдела, Операционный	(null)
Кирилов	3500	Начальник отдела, Продаж	(null)
Абрамов	3700	Начальник отдела, Финансовый	(null)
Березин	3850	Начальник отдела, Административ	(null)
Иванова	2200	Зав. склада	(null)
Пахомова	2000	Зав. склада	(null)
нижкап	2000	Зав. склада	(null)
Кригер	2000	Зав. склада	(null)
Селедцова	2000	Зав. склада	(null)
Жукова	1400	Продавец-консультант	140
Бобков	1490	Продавец-консультант	186,25

# Пример 11

Для предыдущего примера преобразовать неопределенное значение в нулевое. SELECT last\_name, salary, title, COALESCE((salary \* commission\_pct / 100), 0) AS calc\_commission FROM EMP

\$ LAST_NAME		<b>∜</b> TITLE	
Нестеров	5500	Директор	0
Бушуев	3700	Начальник отдела, Операционный	0
Кирилов	3500	Начальник отдела, Продаж	0
Абрамов	3700	Начальник отдела, Финансовый	0
Березин	3850	Начальник отдела, Административ	. 0
Иванова	2200	Зав. склада	0
Пахомова	2000	Зав. склада	0
Пряхин	2000	Зав. склада	0
Кригер	2000	Зав. склада	0
Селедцова	2000	Зав. склада	0
Жукова	1400	Продавец-консультант	140
Бобков	1490	Продавец-консультант	186,25

# Пример 12 12.1 вывести все комбинации номеров отделов и должностей SELECT dept\_id, title FROM EMP

DEPT_ID	↑ TITLE
50	Директор
41	Начальник отдела, Операционный
31	Начальник отдела, Продаж
10	Начальник отдела, Финансовый
50	Начальник отдела, Административный
41	Зав. склада
42	Зав. склада
43	Зав. склада
44	Зав. склада
45	Зав. склада
31	Продавец-консультант
32	Продавец-консультант
33	Продавец-консультант
34	Продавец-консультант
35	Продавец-консультант
41	Служащий
41	Служащий
42	Служащий
42	Служащий
43	Служащий
43	Служащий
44	Служащий
34	Служащий
45	Служащий
45	Служащий

#### 12.2 Вывести без повторений.

SELECT DISTINCT dept\_id, title FROM EMP

\$ DEPT_ID	<b>♦ TITLE</b>
50	Директор
41	Начальник отдела, Операционный
31	Начальник отдела, Продаж
10	Начальник отдела, Финансовый
50	Начальник отдела, Административный
41	Зав. склада
42	Зав. склада
43	Зав. склада
44	Зав. склада
45	Зав. склада
31	Продавец-консультант
32	Продавец-консультант
33	Продавец-консультант
34	Продавец-консультант
35	Продавец-консультант
41	Служащий
42	Служащий
43	Служащий
44	Служащий
34	Служащий
45	Служащий

# Пример 14

вывести из етр фамилии, номера отделов и даты начала работы каждого служащего, результат отсортировать по фамилии.

SELECT last\_name, dept\_id, start\_date FROM EMP ORDER BY last\_name

SEEE T IG	<u> </u>	acpt_1a, 5tt
\$ LAST_NAME	DEPT_ID	\$TART_DATE
Абрамов	10	07.04.90
Белая	33	18.02.91
Бердников	43	26.05.91
Березин	50	04.03.90
Бобков	32	18.01.92
Бушуев	41	08.03.90
Вижров	41	08.03.90
Данилова	45	17.03.91
Дроздов	43	21.07.91
Жукова	31	14.05.90
Иванова	41	18.01.91
Кирилов	31	17.06.91

#### Пример 15

из таблицы етр вывести фамилии, номера отделов, даты найма каждого служащего, результат сортируется так, чтобы служащие, нанятые последними, возглавляли список.

SELECT last\_name, dept\_id, start\_date FROM EMP ORDER BY start\_date DESC

\$ LAST_NAME	DEPT_ID	\$TART_DATE
Кригер	44	09.02.92
Липунова	41	07.02.92
Понаморенко	34	22.01.92
Бобков	32	18.01.92
Кудрявцев	35	09.10.91
Кулик	42	06.08.91
Дроздов	43	21.07.91
Кирилов	31	17.06.91
Бердников	43	26.05.91
Шварц	45	09.05.91
Данилова	45	17.03.91
Селедцова	45	27.02.91
Белая	33	18.02.91
Яшина	42	09.02.91
Иванова	41	18.01.91
Тимофеев	44	30.11.90

пример указания аргумента выражения сортировки не именем столбца, а его порядковым номером в предложении select.

SELECT last\_name, dept\_id, start\_date FROM EMP ORDER BY 2

\$ LAST_NAME		\$TART_DATE
Абрамов	10	07.04.90
Жукова	31	14.05.90
Кирилов	31	17.06.91
Бобков	32	18.01.92
Белая	33	18.02.91
Понаморенко	34	22.01.92
Кулик	34	17.10.90
Кудрявцев	35	09.10.91
Иванова	41	18.01.91
Бушуев	41	08.03.90
Вижров	41	08.03.90
Липунова	41	07.02.92
Пахомова	42	14.05.90

# Пример 17

вывод фамилии, номера отдела, зарплаты всех служащих, результат сортирует по номерам отделов, а в отделах - в порядке убывания зарплаты.

SELECT last\_name, dept\_id, salary FROM EMP ORDER BY dept\_id, salary DESC

\$ LAST_NAME	DEPT_ID	SALARY
Абрамов	10	3700
Кирилов	31	3500
Жукова	31	1400
Бобков	32	1490
Белая	33	1515
Понаморенко	34	1525
Кулик	34	1200
Кудрявцев	35	1450
Бушуев	41	3700
Иванова	41	2200
Липунова	41	1400
Вихров	41	1350
Пахомова	42	2000
Яшина	42	1400
Кулик	42	1200

запрос вывода имен, фамилий, должностей служащих с фамилией Жукова.

SELECT first name, last name, title FROM EMP WHERE last name = 'Жукова'

<pre></pre>	\$ LAST_NAME	<b>∜ TITLE</b>
Светлана	Жукова	Продавец-консультант

#### Пример 19

вывести имена, фамилии и даты найма между 9 мая и 7 июня 1991 включительно.

SELECT first\_name, last\_name, start\_date FROM EMP

WHERE start\_date BETWEEN TO\_DATE('09.05.91', 'DD.MM.YY') AND TO\_DATE('07.06.91', 'DD.MM.YY')

<pre></pre>		\$ START_DATE
Валентин	Бердников	26.05.91
Елизавета	Шварц	09.05.91

#### Пример 20

вывести номера, названия отделов и номера региона для региона 1 и региона 3.

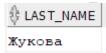
SELECT id, name, region\_id FROM DEPT WHERE region\_id IN (1, 3)

∯ ID	<b>∜</b> NAME	REGION_ID
10	Финансовый	1
31	Продаж	1
33	Продаж	3
41	Операционный	1
43	Операционный	3
50	Административный	1

#### Пример 21

вывести фамилии всех служащих на букву Ж

SELECT last name FROM EMP WHERE last name LIKE 'XK%'



#### Пример 22

вывести фамилии всех служащих, содержащие букву а.

SELECT last\_name FROM EMP WHERE last\_name LIKE '%a%'

<pre></pre>
Абрамов
Иванова
Пахомова
Селедцова
Жукова
Белая
Понаморенко
Липунова
Яшина
Данилова
Шварц

вывести фамилии и даты найма служащих,принятых на работу в 1991 году.

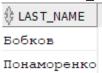
SELECT last\_name, start\_date FROM EMP WHERE TO\_CHAR(start\_date, 'YY') = '91'

\$ LAST_NAME	\$ START_DATE
Кирилов	17.06.91
Иванова	18.01.91
Селедцова	27.02.91
Белая	18.02.91
Кудрявцев	09.10.91
Яшина	09.02.91
Кулик	06.08.91
Дроздов	21.07.91
Бердников	26.05.91
Данилова	17.03.91
Шварц	09.05.91

#### Пример 24

Вывести фамилии, в которых 2я буква о.

SELECT last name FROM EMP WHERE last name LIKE ' o%'



#### Пример 25

вывести названия фирм, содержащие а б.

SELECT name FROM CUSTOMER WHERE name LIKE '%a 6%'



#### Пример 26

вывести номера, наименования и кредитный рейтинг всех клиентов, не имеющих торгового представителя.

SELECT id, name, credit\_rating FROM CUSTOMER WHERE salerep\_id IS NULL

∯ ID	<b>∜</b> NAME	↑ CREDIT_RATING
207	Модуль-К	Хороший

#### Пример 27

вывести фамилии, должности и проценты комиссионных для всех служащих, получающих комиссионные.

SELECT last\_name, title, commission\_pct FROM EMP WHERE commission\_pct IS NOT NULL

\$ LAST_NAME	<b>∜ TITLE</b>	⊕ COMMISSION_PCT
Жукова	Продавец-консультант	10
Бобков	Продавец-консультант	12,5
Белая	Продавец-консультант	10
Понаморенко	Продавец-консультант	15
Кудрявцев	Продавец-консультант	17,5

вывести фамилии, зарплаты и номера отделов для работников склада отдела №41.

SELECT last\_name, salary, dept\_id FROM EMP WHERE dept\_id = 41

\$ LAST_NAME	\$ SALARY	
Бушуев	3700	41
Иванова	2200	41
Липунова	1400	41
Вихров	1350	41

#### Пример 29

вывести фамилии, зарплаты и номера отдела для всех служащих, являющихся работниками склада или сотрудниками отдела №41.

SELECT last\_name, salary, dept\_id FROM EMP WHERE title LIKE '% склада%' OR dept\_id = 41

\$ LAST_NAME	\$ SALARY	DEPT_ID
Бушуев	3700	41
Иванова	2200	41
Пахомова	2000	42
Пряхин	2000	43
Кригер	2000	44
Селедцова	2000	45
Липунова	1400	41
Вихров	1350	41

#### Пример 30

вывод информации о служащих 44 отдела, зарплата которых составляет 1000 и более, а также о всех служащих отдела 42 (фамилия, зарплата, номер отдела).

SELECT last\_name, salary, dept\_id FROM EMP WHERE (dept\_id = 44 AND salary >= 1000) OR dept\_id = 42

\$ LAST_NAME	\$ SALARY	
Пахомова	2000	42
Кригер	2000	44
Яшина	1400	42
Кулик	1200	42
Тимофеев	1350	44

#### Пример 31

вывести фамилию, зарплату и номера отделов для всех служащих отдела 44 и 42 и зарабатывающих 1000 и более.

SELECT last\_name, salary, dept\_id FROM EMP WHERE dept\_id IN (42, 44) AND salary >= 1000

\$ LAST_NAME	\$ SALARY	
Пахомова	2000	42
Кригер	2000	44
Яшина	1400	42
Кулик	1200	42
Тимофеев	1350	44

#### Пример 32

вывести имена и фамилии вице-президентов строчными буквами, идентификатора пользователя с заглавной буквы и должности заглавными буквами.

SELECT LOWER(first\_name) AS first\_name, LOWER(last\_name) AS last\_name, INIT-CAP(userid) AS userid, UPPER(title) AS title

#### FROM EMP WHERE title LIKE '%Начальник отдела%'

\$ FIRST_NAME	\$ LAST_NAME		<b>∜ TITLE</b>		
леонид	бушуев	лбушуев	начальник	отдела,	операционный
михаил	кирилов	Мкирилов	начальник	отдела,	продаж
виктор	абрамов	Вабрамов	начальник	отдела,	ФИНАНСОВЫЙ
николай	березин	Нберезин	начальник	отдела,	административный

#### Пример 33

вывести имена и фамилии всех служащих с фамилией Кулик.

SELECT first name, last name FROM EMP WHERE last name = 'Кулик'

FIRST_NAME	\$ LAST_NAME
Евгений	Кулик
Ирина	Кулик

#### Пример 34

вывести наименование и страны всех клиентов с хорошим кредитным рейтингом, причем наименование и страна должны быть соединены (между ними нет пробела).

SELECT name || country AS Name\_and\_country FROM CUSTOMER WHERE credit\_rating = 'Хороший'

\$ NAME_AND_COUNTRY		
Алтай сибстройРоссия		
Модуль-КРоссия		
ЛакмусРоссия		

#### Пример 35

вывести все наименования товаров, первые буквы которых равны BNC, и длины этих наименований.

SELECT name, LENGTH(name) AS Length FROM PRODUCT WHERE name LIKE 'BNC%'



#### Пример 36

обеспечить вывод значения числа 45.923 с округлением до сотых и без округления, и с округлением до десятых.

SELECT ROUND(45.923, 2) AS rounded\_to\_hundredths, 45.923 AS no\_rounding,

ROUND(45.923, 1) AS rounded\_to\_tenths FROM DUAL

♠ ROUNDED_TO_HUNDREDTHS	♦ NO_ROUNDING	₱ ROUNDED_TO_TENTHS
45,92	45,923	45,9

# Пример 37

вывести 45.923 с усечением до сотых, без усечения, и с усечением до десятых.

SELECT TRUNC(45.923, 2) AS truncated\_to\_hundredths, 45.923 AS no\_truncation,

TRUNC(45.923, 1) AS truncated\_to\_tenths FROM DUAL

# TRUNCATED_TO_HUNDREDTHS	\$ NO_TRUNCATION	# TRUNCATED_TO_TENTHS
45,92	45,923	45,9

#### Пример 38

вывести остаток от деления зарплаты на комиссионные для служащих с зарплатой выше 1400.

SELECT last\_name, MOD(salary, commission\_pct) AS remainder FROM EMP WHERE salary > 1400

<pre></pre>	REMAINDER
Нестеров	(null)
Бушуев	(null)
Кирилов	(null)
Абрамов	(null)
Березин	(null)
Иванова	(null)
Пахомова	(null)
нижкеп	(null)
Кригер	(null)
Селедцова	(null)
Бобков	2,5
Белая	5
Понаморенко	10
Кудрявцев	15
Дроздов	(null)
Данилова	(null)

вывести текущую дату с помощью таблицы DUAL.

SELECT SYSDATE FROM DUAL

\$ SYSDATE

#### Пример 40

вывести фамилии и кол-во отработанных недель для служащих 43 отдела.

SELECT last\_name, TRUNC(ABS(SYSDATE - start\_date) / 7, 0) AS weeks\_worked FROM EMP WHERE dept\_id = 43

\$ LAST_NAME	
Пряхин	3453
Дроздов	3520
Бердников	3512

# Пример 41

вывести номер служащего, кол-во отработанных месяцев и даты полугодовой аттестации для служащих работающих менее 48 месяцев.

 $SELECT\ id,\ ABS(MONTHS\_BETWEEN(SYSDATE,\ start\_date))\ AS\ months\_worked,$ 

ADD\_MONTHS(start\_date, 6) AS next\_evaluation FROM EMP

WHERE ABS(MONTHS\_BETWEEN(SYSDATE, start\_date)) < 48

#### Пример 42

вывести в хронологическом порядке номера каждого товара, запас которого пополнился, первой пятницы после даты пополнения запаса и последнего дня месяца, когда пополнился запас.

SELECT ITEM.product\_id,

ORD.date\_shipped,

NEXT DAY(ORD.date shipped, 'ПЯТНИЦА') AS first friday after shipment,

LAST\_DAY(ORD.date\_shipped) AS last\_day\_of\_shipment\_month

FROM ITEM

JOIN ORD ON ITEM.ord\_id = ORD.id

ORDER BY ORD.date\_shipped

PRODUCT_ID	↑ DATE_SHIPPED		\$\text{LAST_DAY_OF_SHIPMENT_MONTH}
20108	08.09.92	12.09.92	30.09.92
20201	08.09.92	12.09.92	30.09.92
10021	10.09.92	12.09.92	30.09.92
10013	10.09.92	12.09.92	30.09.92
10011	10.09.92	12.09.92	30.09.92
20106	10.09.92	12.09.92	30.09.92
20510	10.09.92	12.09.92	30.09.92
41080	10.09.92	12.09.92	30.09.92
41100	10.09.92	12.09.92	30.09.92
32861	10.09.92	12.09.92	30.09.92
20512	10.09.92	12.09.92	30.09.92
32779	10.09.92	12.09.92	30.09.92
30433	10.09.92	12.09.92	30.09.92
10023	10.09.92	12.09.92	30.09.92
40421	10.09.92	12.09.92	30.09.92
30321	10.09.92	12.09.92	30.09.92
41010	10.09.92	12.09.92	30.09.92
30326	10.09.92	12.09.92	30.09.92
20201	15.09.92	19.09.92	30.09.92
50419	15.09.92	19.09.92	30.09.92

сравнить даты найма всех служащих, нанятых в 1991 году, вывести номера каждого служащего, даты начала и месяца начала работы с помощью функций TRUNC и ROUND.

SELECT id, start\_date,

EXTRACT(MONTH FROM TO\_DATE(start\_date, 'DD.MM.YY')) AS start\_month FROM EMP

WHERE TO\_CHAR(start\_date, 'YY') = '91'

ORDER BY start\_date

∯ ID	\$ START_DATE	<pre></pre>
6	18.01.91	1
18	09.02.91	2
13	18.02.91	2
10	27.02.91	2
24	17.03.91	3
25	09.05.91	5
21	26.05.91	5
3	17.06.91	6
20	21.07.91	7
19	06.08.91	8
15	09.10.91	10

#### Пример 44

вывести номер и дату каждого заказа, принятого торговым представителем №11, причем вывод дат должен производиться в виде 08/92.

SELECT id AS order\_id, TO\_CHAR(date\_ordered, 'MM/YY') AS order\_date FROM ORD

WHERE salerep\_id = 11

ORDER_ID	\$ ORDER_DATE
100	08/92
105	09/92
109	09/92
110	09/92
111	09/92

вывести подзаголовком "Начальник отдела" фамилию каждого начальника заглавными буквами и части названия отделов.

SELECT UPPER(last\_name) AS chief\_last\_name, SUBSTR(title, 19) AS department\_name FROM EMP

WHERE title LIKE 'Начальник отдела%'

	, ,
CHIEF_LAST_NAME	DEPARTMENT_NAME
БУШУЕВ	Операционный
кирилов	Продаж
АБРАМОВ	Финансовый
БЕРЕЗИН	Административный

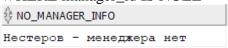
#### Пример 46

вывести фамилию руководителя фирмы, у которого нет менеджера, выходная строка должна указывать, что для этой фамилии менеджера нет.

SELECT last\_name || ' - менеджера нет' AS no\_manager\_info

FROM EMP

WHERE manager\_id IS NULL



#### Пример 47

сделать соединение таблиц служащих и отделов для вывода фамилии служащего, номера и названия отдела.

SELECT EMP.last\_name AS employee\_last\_name,

DEPT.id AS dept\_id,

DEPT.name AS dept\_name

FROM EMP

JOIN DEPT ON EMP.dept\_id = DEPT.id

⊕ EMPLOYEE_LAST_NAME	∯ DEPT ID	⊕ DEPT_NAME
Нестеров		Административный
Бушуев		Операционный
Кирилов	31	Продаж
Абрамов	10	Финансовый
Березин	50	Административный
Иванова	41	Операционный
Пахомова	42	Операционный
нижедП	43	Операционный
Кригер	44	Операционный
Селедцова	45	Операционный
Жукова	31	Продаж
Бобков	32	Продаж
Белая	33	Продаж

#### Пример 48

вывести номера отдела, номера региона и название региона для всех отделов.

SELECT DEPT.id AS dept\_id,

DEPT.region\_id,

REGION.name AS region\_name

FROM DEPT

JOIN REGION ON DEPT.region\_id = REGION.id

\$ DEPT_ID	REGION_ID	
10	1	Новосибирская обл
31	1	Новосибирская обл
32	2	Омская обл
33	3	Кемеровская обл
34	4	Алтайский край
35	5	Красноярский край
41	1	Новосибирская обл
42	2	Омская обл

вывести фамилию, номер отдела, название отдела сотрудника с фамилией Бердников SELECT EMP.last\_name,

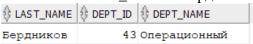
EMP.dept\_id,

DEPT.name AS dept\_name

FROM EMP

JOIN DEPT ON EMP.dept\_id = DEPT.id

WHERE EMP.last\_name = 'Бердников'



#### Пример 51

вывести наименование клиента, номера региона и его название для всех клиентов. Использовать псевдонимы столбцов и таблиц.

SELECT C.name AS customer\_name,

R.id AS region\_id,

R.name AS region\_name

FROM CUSTOMER C

JOIN REGION R ON C.region\_id = R.id

	# REGION_ID	REGION_NAME
Омский транспорт	2	Омская обл
Риом	4	Алтайский край
Алтай сибстрой	4	Алтайский край
Бизнес-Практика	1	Новосибирская обл
Ромул	4	Алтайский край
Сфера	5	Красноярский край
Модуль-К	3	Кемеровская обл
Лакмус	5	Красноярский край
Бриг 95	1	Новосибирская обл
ОмГУПС	2	Омская обл

#### Пример 52

создать не эквисоединение для вычисления категории служащего по зарплате, которая должна быть между любой парой нижнего и верхнего значения диапазона зарплат.

```
CREATE\ TABLE\ SALARY\_RANGES (
```

```
id INT PRIMARY KEY,
min_salary DECIMAL(10, 2),
max_salary DECIMAL(10, 2),
salary_category VARCHAR(20)
```

#### **INSERT ALL**

)

INTO SALARY\_RANGES (id, min\_salary, max\_salary, salary\_category) VALUES (1, 1250, 2500, 'Низкая')

INTO SALARY\_RANGES (id, min\_salary, max\_salary, salary\_category) VALUES (2, 2501, 4000, 'Средняя')

INTO SALARY\_RANGES (id, min\_salary, max\_salary, salary\_category) VALUES (3, 4001, 5500, 'Высокая')

SELECT \* FROM dual

#### SELECT \* FROM SALARY\_RANGES

SELECT EMP.last\_name, EMP.salary, SR.salary\_category FROM EMP

JOIN SALARY\_RANGES SR ON EMP.salary >= SR.min\_salary AND EMP.salary <=

SR.max\_salary

∯ ID ‡	MIN_SALARY	MAX_SALARY	\$ SALARY_CATEGORY
1	1250	2500	Низкая
2	2501	4000	Средняя
3	4001	5500	Высокая

\$ LAST_NAME		
Пряхин	2000	Низкая
Кригер	2000	Низкая
Селедцова	2000	Низкая
Жукова	1400	Низкая
Бобков	1490	Низкая
Белая	1515	Низкая
Понаморенко	1525	Низкая
Кудрявцев	1450	Низкая
Липунова	1400	Низкая
Вижров	1350	Низкая
Яшина	1400	Низкая
Дроздов	1450	Низкая
Тимофеев	1350	Низкая
Данилова	1450	Низкая
Шварц	1250	Низкая
Бушуев	3700	Средняя
Кирилов	3500	Средняя
Абрамов	3700	Средняя
Березин	3850	Средняя
Нестеров	5500	Высокая

#### Пример 53

вывести для каждого клиента его наименование, фамилию и идентификационный номер торгового представителя, в список включены так же клиенты, у которых нет торговых представителей.

SELECT CUSTOMER.name AS customer\_name,

EMP.last\_name,

CUSTOMER.salerep\_id

FROM CUSTOMER

LEFT JOIN EMP ON CUSTOMER.salerep\_id = EMP.id

	<pre></pre>	<pre></pre>
Бизнес-Практика	Жукова	11
Бриг 95	Жукова	11
Иподор	Жукова	11
Галинское АОЗТ	Жукова	11
Омский трансп	Бобков	12
ОмГУПС	Бобков	12
Монолит	Белая	13
Риом	Понаморенко	14
Алтай сибстрой	Понаморенко	14
Ромул	Кудрявцев	15
Сфера	Кудрявцев	15
Лакмус	Кудрявцев	15
000 Талан	Кудрявцев	15
Эпекли	Кудрявцев	15
Модуль-К	(null)	(null)

вывести имена сотрудников и их менеджеров.

SELECT E.first\_name AS employee\_name,

M.first\_name AS manager\_name

FROM EMP E

LEFT JOIN EMP M ON E.manager\_id = M.id

\$ EMPLOYEE_NAME	
Леонид	Виктор
Михаил	Виктор
Виктор	Виктор
Николай	Виктор
Нина	Леонид
Нина	Леонид
Владимир	Леонид
Людмила	Леонид
Людмила	Леонид
Светлана	Михаил
Михаил	Михаил
Галина	Михаил
Надежда	Михаил
Андрей	Михаил
Елена	Нина

#### Пример 55

вывод средней, самой высокой и низкой зарплаты, суммы зарплаты всех торговых представителей за месяц.

SELECT AVG(salary) AS avg\_salary,

MAX(salary) AS max\_salary,

MIN(salary) AS min\_salary,

SUM(salary) AS total\_salary

#### FROM EMP

# AVG_SALARY	# MAX_SALARY	∯ MIN_SALARY	
2043,2	5500	1200	51080

#### Пример 56

вывод первой и последней фамилии из алфавитного списка всех служащих.

SELECT MIN(last\_name) AS first\_last\_name,

MAX(last\_name) AS last\_last\_name

FROM EMP

<pre></pre>	\$ LAST_LAST_NAME
Абрамов	Яшина

вывод всех сотрудников отдела 31.

SELECT id, last\_name, first\_name, dept\_id FROM EMP WHERE dept\_id = 31

∯ ID	↓ LAST_NAME	<pre></pre>	DEPT_ID
3	Кирилов	Михаил	31
11	Жукова	Светлана	31

#### Пример 58

вывод количества сотрудников отдела 31, имеющих комиссионные.

SELECT COUNT(\*) AS count

FROM EMP

WHERE dept\_id = 31 AND commission\_pct IS NOT NULL



#### Пример 59

вывод всех возможных кредитных рейтингов и количество клиентов в каждой категории, причем столбец должен иметь заголовок "#Cust".

SELECT credit\_rating, COUNT(\*) AS "#Cust"

FROM CUSTOMER

GROUP BY credit\_rating

\$ CREDIT_RATING	<pre>#Cust</pre>
Отличный	9
Плохой	3
Хороший	3

#### Пример 60

вывод всех должностей, кроме начальников отделов и соответствующей общей зарплаты за месяц, список сортируется по общей зарплате.

SELECT title, SUM(salary) AS total\_salary

FROM EMP

WHERE title NOT LIKE 'Начальник отдела%'

**GROUP BY title** 

ORDER BY total\_salary DESC

↑ TITLE	
Служащий	13250
Зав. склада	10200
Продавец-консультант	7380
Директор	5500

#### Пример 61

вывод максимальной зарплаты для каждой должности без указания самой должности. SELECT MAX(salary)

FROM EMP

**GROUP BY title** 

\$ MAX(SALARY)	
	5500
	3700
	3500
	3700
	3850
	2200
	1525
	1450

вывод максимальной зарплаты для каждой должности с указанием самой должности.

SELECT title, MAX(salary) AS max\_salary

FROM EMP

**GROUP BY title** 

Директор	5500
Начальник отдела, Операционный	3700
Начальник отдела, Продаж	3500
Начальник отдела, Финансовый	3700
Начальник отдела, Административ	. 3850
Зав. склада	2200
Продавец-консультант	1525
Служащий	1450

#### Пример 63

выбрать номера регионов (их кол-во) с определенным номером.

select region\_id, COUNT(name) from dept;

неправильно, не указано поле группировки.

ORA-00937: групповая функция не является одногруппной 00937. 00000 - "not a single-group group function" \*Cause: \*Action: Error at Line: 1 Column: 8

# Пример 64

правильно

select region\_id, COUNT(name) from dept GROUP BY region\_id;

REGION_ID	DEPT_COUNT
1	4
2	2
3	2
4	2
5	2

#### Пример 65

вывести номера отделов, зарплаты для отделов, где средняя зарплата превышает 2000.

SELECT dept\_id, AVG(salary) AS avg\_salary

FROM EMP

GROUP BY dept id

HAVING AVG(salary) > 2000

\$ DEPT_ID	AVG_SALARY
50	4675
41	2162,5
31	2450
10	3700

вывод количества служащих по должностям внутри отделов.

SELECT dept\_id, title, COUNT(\*) AS emp\_count

FROM EMP

GROUP BY dept\_id, title

DEPT_ID	↑ TITLE	
50	Директор	1
41	Начальник отдела, Операционный	1
31	Начальник отдела, Продаж	1
10	Начальник отдела, Финансовый	1
50	Начальник отдела, Административ	. 1
41	Зав. склада	1
42	Зав. склада	1
43	Зав. склада	1
44	Зав. склада	1
45	Зав. склада	1
31	Продавец-консультант	1
32	Продавец-консультант	1
33	Продавец-консультант	1
34	Продавец-консультант	1
35	Продавец-консультант	1
41	Служащий	2

#### Пример 67

вывод количества служащих в каждой должности по отделам.

#### Пример 68

вывод должности общей зарплаты для всех должностей с зарплатой больше 1350 в месяц, кроме начальников отделов, выходные строки сортируются по зарплате.

SELECT title, SUM(salary) AS total\_salary

FROM EMP

WHERE salary > 1350 AND title NOT LIKE 'Начальник отдела%'

GROUP BY title

ORDER BY total\_salary DESC

∯ TITLE	
Зав. склада	10200
Продавец-консультант	7380
Служащий	5700
Директор	5500

#### Пример 69

вывод всех номеров отделов с общей зарплатой более 2500 в месяц.

SELECT dept\_id, SUM(salary) AS total\_salary

FROM EMP

HAVING SUM(salary) > 2500

GROUP BY dept\_id

DEPT_ID	\$ TOTAL_SALARY
50	9350
41	8650
31	4900
10	3700
42	4600
43	4650
44	3350
45	4700
34	2725

выборка фамилий и должностей сотрудников того отдела, где работает Кирилов.

SELECT E.last\_name, E.title

FROM EMP E

JOIN EMP R ON E.dept\_id = R.dept\_id AND R.last\_name = 'Кирилов'

\$ LAST_NAME	∯ TITLE		
Кирилов	Начальник отдела, Пр	родаж	
Жукова	Продавец-консультант		

#### Пример 71

вывод фамилий сотрудников, имеющих такую же должность как у Абрамова.

```
SELECT EMP.last_name
FROM EMP
WHERE EMP.title = (
    SELECT title
    FROM EMP
    WHERE last_name = 'ΑδραμοΒ'
)
AND EMP.last_name != 'ΑδραμοΒ'

LAST_NAME
```

#### Пример 72

вывод фамилий, должностей, зарплат всех сотрудников с зарплатой ниже среднего.

SELECT last\_name, title, salary

FROM EMP

WHERE salary < (SELECT AVG(salary) FROM EMP)

\$ LAST_NAME	∜ TITLE	∯ SALARY
Пахомова	Зав. склада	2000
Пряжин	Зав. склада	2000
Кригер	Зав. склада	2000
Селедцова	Зав. склада	2000
Жукова	Продавец-консульт	1400
Бобков	Продавец-консульт	1490
Белая	Продавец-консульт	1515
Понаморенко	Продавец-консульт	1525
Кудрявцев	Продавец-консульт	1450
Липунова	Служащий	1400

## Пример 73

вывод списка служащих, принадлежащих к финансовому отделу или ко второму региону.

SELECT EMP.last name, EMP.title, DEPT.name, DEPT.region id

FROM EMP

JOIN DEPT ON EMP.dept id = DEPT.id

WHERE EMP.title = 'Служащий' AND (DEPT.name = 'Финансовый' OR DEPT.region\_id = 2)

\$ LAST_NAME	↑ TITLE	NAME	REGION_ID
Яшина	Служащий	Операционный	2
Кулик	Служащий	Операционный	2

#### Пример 74

вывести все отделы, где зарплата выше, чем в отделе 32.

SELECT DEPT.id, DEPT.name

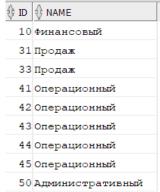
FROM DEPT

WHERE (SELECT AVG(EMP.salary)

FROM EMP

WHERE EMP.dept\_id = DEPT.id) > (SELECT AVG(EMP.salary) FROM EMP

WHERE EMP.dept\_id = 32)



#### Пример 75

поиск должности с самой низкой средней зарплатой.

SELECT title, AVG(salary) AS avg\_salary

FROM EMP

**GROUP BY title** 

ORDER BY avg salary ASC

FETCH FIRST 1 ROW ONLY



#### Пример 76

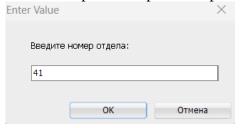
создать команду, которая во время выполнения запроса запрашивает номер отдела сотрудника, отчет должен содержать учетный номер, фамилию и зарплату.

ACCEPT param\_dept\_id CHAR PROMPT 'Введите номер отдела: '

SELECT id, last\_name, salary

FROM EMP

WHERE dept\_id = '&param\_dept\_id'



2 Бушуев	3700
6 Иванова	2200
16 Липунова	1400
17 Вихров	1350

#### Пример 77

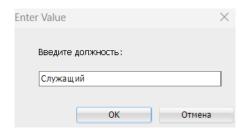
выбрать номера, фамилии и зарплаты служащих, находящихся на конкретной должности, должность запросить во время выполнения.

ACCEPT param\_title CHAR PROMPT 'Введите должность: '

SELECT id, last\_name, salary

FROM EMP

WHERE title = '&param\_title'





обеспечить вывод номера заказа любого другого столбца или любого условия заказа (можно попробовать разные условия и имена столбцов и сравнить результаты).

#### Пример 79

вывести номера и названия региона для заданного отдела (надо создать командный файл с именем masha.sql).

ACCEPT param\_dept\_id CHAR PROMPT 'Введите номер отдела: '

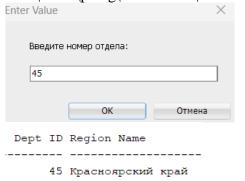
SELECT DEPT.id AS "Dept ID", REGION.name AS "Region Name"

FROM DEPT

JOIN REGION ON DEPT.region\_id = REGION.id

WHERE DEPT.id = '&param dept id'

#### @C:\Users\pereg\Downloads\masha.sql



#### Пример 80

создать переменную для хранения позиции отдел и обеспечить вывод пользователя всех отделов, которые соответствуют данной переменной.

#### Пример 81

создать командный файл с именем masha2.sql для получения серии отчетов по должностям, должности вводить в командной строке после имени скрипта.

SELECT id, last\_name, salary

FROM EMP

WHERE title = '&1'

#### @C:\Users\pereg\Downloads\masha2.sql "Зав. склада"

ID	LAST_NAME	SALARY
6	Иванова	2200
7	Пахомова	2000
8	Пряжин	2000
9	Кригер	2000
10	Селедцова	2000

#### Пример 82

ограничение NOT NULL задано для столбца PHONE, который содержит символьные строки переменной длины не более 14 символов (имя ограничения сервер создаст сам).

```
CREATE TABLE EmployeesPhones (
ID NUMBER PRIMARY KEY,
Name VARCHAR2(100),
PHONE VARCHAR2(14) NOT NULL
```

#### Пример 83

ограничение задано для столбца last\_name, который содержит строки переменной длины не более 25 символов.

#### Создание таблицы.

- создать командный файл (скрипт) начиная с команды CREATE TABLE и указать имя таблины.
- указать имена столбцов, типы данных и их длину, определение столбцов разбить запятыми.
- задать ограничение NOT NULL на уровне столбцов, кроме столбцов связанных с ограниченеим PK.
- задать ограничения РК на уровне столбца, если оно оносится только к одному столбцу, или на уровне таблицы, если оно относится к нескольким столбцам.
- задать ограничения UNIQUE, CHECK и FK.
- сохранить и вполнить командный файл.

```
CREATE TABLE EmployeesInfo (
ID NUMBER CONSTRAINT emp_info_id_pk PRIMARY KEY,
Last_Name VARCHAR2(25) NOT NULL,
Phone VARCHAR2(14),
Department_ID NUMBER,
CONSTRAINT emp_info_phone_unique UNIQUE (Phone),
CONSTRAINT emp_info_dept_fk FOREIGN KEY (Department_ID) REFERENCES Departments(ID),
CONSTRAINT emp_last_name_check CHECK (Last_Name != ")
)

CREATE TABLE Departments (
ID NUMBER PRIMARY KEY,
Name VARCHAR2(50)
)
```

```
CREATE TABLE EmployeesInfo (

ID NUMBER CONSTRAINT emp_info_id_pk PRIMARY KEY,
Last_Name VARCHAR2(25) NOT NULL,
Phone VARCHAR2(14),
Department_ID NUMBER,
CONSTRAINT emp_info_phone_unique UNIQUE (Phone),
CONSTRAINT emp_info_dept_fk FOREIGN KEY (Department_ID)
REFERENCES Departments(ID),
CONSTRAINT emp_last_name_check CHECK (Last_Name != '')
)

Table EMPLOYEESINFO created.
```

#### Пример 84.

region\_id);

create table s\_dept (id NUMBER(7) constraint s\_dept\_id\_pk PRIMARY KEY, name VARCHAR2(25) constraint s\_dept\_name\_nn NOT NULL, region\_id NUMBER(7) constraint s\_dept\_region\_id\_fk REFERENCES s\_region(id), constraint s\_dept\_name\_region\_id\_uk UNIQUE (name, region\_id)); где ограничение s dept id pk задается на уровне столбца и делает столбец id PK;

ограничение s\_dept\_name\_nn на уровне столбца, означает, что каждый отдел в таблице s\_dept имеет имя.

s\_dept\_region\_fk - на уровне столбца, значит, что любой номер региона, введенный в таблицу s\_dept имеет соответствующее значение в талице s\_region, прежде чем задать ограничение надо создать таблицу s\_region, а так же ограничение UNIQUE для столбца id.

s\_dept\_name\_region\_id\_uk на уровне таблицы определяет столбцы name и region\_id как составной уникальный ключ, чтобы ни одна комбинация имени региона и номера региона не появилась >1 раза.

Можно создать таблицу заполняя строками другой таблицы.

```
Синтаксис:
create table таблица [столбец (, столбец ...)] AS подзапрос;
где таблица - имя таблицы;
  столбец - имя столбца, значение по умолчанию и ограничение целостности.
  подзапрос - команда select, определяющая строки для вставки в новую таблицу.
CREATE TABLE s_region (
  id NUMBER(7) CONSTRAINT s_region_id_pk PRIMARY KEY,
  name VARCHAR2(50) CONSTRAINT s_region_name_nn NOT NULL
)
CREATE TABLE s_dept AS
SELECT id, name, region id
FROM DEPT
WHERE 1=0
ALTER TABLE s_dept ADD CONSTRAINT s_dept_id_pk PRIMARY KEY (id);
ALTER TABLE s_dept ADD CONSTRAINT s_dept_region_id_fk FOREIGN KEY (region_id)
REFERENCES s_region(id);
ALTER TABLE s_dept ADD CONSTRAINT s_dept_name_region_id_uk UNIQUE (name,
```

```
Table S_REGION created.

Table S_DEPT created.

Table S_DEPT altered.

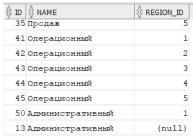
Table S_DEPT altered.
```

ввести новый отдел без номера региона, отсутствие номера региона INSERT означает неявный ввод неопределенного значения для этого столбца в новой строке. INSERT INTO DEPT VALUES (13, 'Административный', NULL)

#### Пример 86

insert into dept values(13, 'Административный', NULL);

1 row inserted.



#### Пример 87

перевод служащего 2 в отдел 10 и служащего 1 в отдел 32 и повышение его зарплаты на 2500.

UPDATE EMP SET dept\_id = 10 WHERE id = 2; UPDATE EMP SET dept\_id = 32, salary = salary + 2500 WHERE id = 1

1 row updated.

1 row updated.

#### Пример 88

update emp set dept\_id=32, salary=2500 where id=1;

#### Пример 89

удаление информации о служащих, принятых на работу после января 1996 года. DELETE FROM EMP WHERE start\_date >= TO\_DATE('01.02.96', 'DD.MM.YY')

#### Пример 90.

удалить все данные таблицы test.

**DELETE FROM TEST**