

Задания:

1. Выполнить запрос, выводящий все строки из таблицы Employees.
2. Выполнить запрос, который выведет отсортированные по первой колонке данные таблицы Employees.
3. Выведите имя, фамилию, email, телефон сотрудников, которых зовут Peter.
4. Выведите имя, фамилию и идентификатор сотрудников, у которых зарплата более \$6000.
5. Выведите имя, фамилию и зарплату сотрудников, которые получают больше, чем \$2500 и меньше, чем \$4000.
6. Выведите имя, фамилию, зарплату в долларах и зарплату в рублях (можно выбрать валюту своей страны) по текущему курсу для тех сотрудников, которые получают более 5000 рублей (считайте, что в таблице зарплата указана в долларах).
7. Выведите все адреса из Великобритании с указанием отделов, которые по этим адресам находятся. (связь таблиц Countries, Locations).
8. Выведите все страны, название которых содержит букву "I". Вывести страну и город.
9. Выведите менеджеров для всех отделов кроме 10, 60, 80, 120.

1. Выполнить запрос, выводящий все строки из таблицы Employees.

Запрос: SELECT * FROM HR.employees;

Welcome Page XEPDB1											
Worksheet Query Builder											
SELECT * FROM HR.employees;											
Query Result x											
All Rows Fetched: 107 in 0.065 seconds											
EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID	
1	100 Steven	King	SKING	515.123.4567	17-JUN-03	AD_PRES	24000	(null)	(null)	90	
2	101 Neena	Kochhar	NKOCHHAR	515.123.4568	21-SEP-05	AD_VP	17000	(null)	100	90	
3	102 Lex	De Haan	LDEHAAN	515.123.4569	13-JAN-01	AD_VP	17000	(null)	100	90	
4	103 Alexander	Hunold	AHUNOLD	590.423.4567	03-JAN-06	IT_PROG	9000	(null)	102	60	
5	104 Bruce	Ernst	BERNST	590.423.4568	21-MAY-07	IT_PROG	6000	(null)	103	60	
6	105 David	Austin	DAUSTIN	590.423.4569	25-JUN-05	IT_PROG	4800	(null)	103	60	
7	106 Valli	Pataballa	VPATABAL	590.423.4560	05-FEB-06	IT_PROG	4800	(null)	103	60	
8	107 Diana	Lorentz	DLORENTZ	590.423.5567	07-FEB-07	IT_PROG	4200	(null)	103	60	
9	108 Nancy	Greenberg	NGREENBE	515.124.4569	17-AUG-02	FI_MGR	12008	(null)	101	100	
10	109 Daniel	Faviet	DFAVIET	515.124.4169	16-AUG-02	FI_ACCOUNT	9000	(null)	108	100	
11	110 John	Chen	JCHEN	515.124.4269	28-SEP-05	FI_ACCOUNT	8200	(null)	108	100	
12	111 Tasmal	Sciarra	ISCIARRA	515.124.4369	30-SEP-05	FI_ACCOUNT	7700	(null)	108	100	
13	112 Jose Manuel	Urman	JMURMAN	515.124.4469	07-MAR-06	FI_ACCOUNT	7800	(null)	108	100	
14	113 Luis	Popp	LPOPP	515.124.4567	07-DEC-07	FI_ACCOUNT	6900	(null)	108	100	
15	114 Den	Raphaely	DRAPHEAL	515.127.4561	07-DEC-02	FU_MAN	11000	(null)	100	30	
16	115 Alexander	Khoo	AKHOO	515.127.4562	18-MAY-03	FU_CLERK	3100	(null)	114	30	
17	116 Shelli	Baida	SBAIDA	515.127.4563	24-DEC-05	FU_CLERK	2900	(null)	114	30	
18	117 Sigal	Tobias	STOBIAS	515.127.4564	24-JUL-05	FU_CLERK	2800	(null)	114	30	
19	118 Guy	Himuro	GHIMURO	515.127.4565	15-NOV-06	FU_CLERK	2600	(null)	114	30	
20	119 Karen	Colmenares	KCOLMENA	515.127.4566	10-AUG-07	FU_CLERK	2500	(null)	114	30	
21	120 Matthew	Weiss	MWEISS	650.123.1234	18-JUL-04	ST_MAN	8000	(null)	100	50	
22	121 Adam	Fripp	AFRIPP	650.123.2234	10-APR-05	ST_MAN	8200	(null)	100	50	
23	122 Payam	Kaufling	PKAUFLIN	650.123.3234	01-MAY-03	ST_MAN	7900	(null)	100	50	
24	123 Shanta	Vollman	SVOLLMAN	650.123.4234	10-OCT-05	ST_MAN	6500	(null)	100	50	
25	124 Kevin	Mourgos	KMOURGOS	650.123.5234	16-NOV-07	ST_MAN	5800	(null)	100	50	
26	125 Julia	Nayer	JNAYER	650.124.1214	16-JUL-05	ST_CLERK	3200	(null)	120	50	
27	126 Irene	Mikkilineni	IMIKKILI	650.124.1224	28-SEP-06	ST_CLERK	2700	(null)	120	50	
28	127 James	Landry	JLANDRY	650.124.1334	14-JAN-07	ST_CLERK	2400	(null)	120	50	
29	128 Steven	Markle	SMARKLE	650.124.1434	08-MAR-08	ST_CLERK	2200	(null)	120	50	
30	129 Laura	Bissot	LBISSOT	650.124.5234	20-AUG-05	ST_CLERK	3300	(null)	121	50	
31	130 Mozhe	Atkinson	MATKINSO	650.124.6234	30-OCT-05	ST_CLERK	2800	(null)	121	50	
32	131 James	Marlow	JAMLOW	650.124.7234	16-FEB-05	ST_CLERK	2500	(null)	121	50	
33	132 TJ	Olson	TJOLSON	650.124.8234	10-APR-07	ST_CLERK	2100	(null)	121	50	
34	133 Jason	Mallin	JMALLIN	650.127.1934	14-JUN-04	ST_CLERK	3300	(null)	122	50	

2. Выполнить запрос, который выведет отсортированные по первой колонке данные таблицы Employees.

Запрос: SELECT * FROM HR.employees ORDER BY 1;

Worksheet		Query Builder									
<pre>SELECT * FROM HR.employees ORDER BY 1;</pre>											
Query Result x											
All Rows Fetched: 107 in 0.011 seconds											
	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
1	100	Steven	King	SKING	515.123.4567	17-JUN-03	AD_PRES	24000	(null)	(null)	90
2	101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-SEP-05	AD_VP	17000	(null)	100	90
3	102	Lex	De Haan	LDEHAAN	515.123.4569	13-JAN-01	AD_VP	17000	(null)	100	90
4	103	Alexander	Hunold	AHUNOLD	590.423.4567	03-JAN-06	IT_PROG	9000	(null)	102	60
5	104	Bruce	Ernst	BERNST	590.423.4568	21-MAY-07	IT_PROG	6000	(null)	103	60
6	105	David	Austin	DAUSTIN	590.423.4569	25-JUN-05	IT_PROG	4800	(null)	103	60
7	106	Valli	Pataballa	VPATABAL	590.423.4560	05-FEB-06	IT_PROG	4800	(null)	103	60
8	107	Diana	Lorentz	DLORENTZ	590.423.5567	07-FEB-07	IT_PROG	4200	(null)	103	60
9	108	Nancy	Greenberg	NGREENBE	515.124.4569	17-AUG-02	FI_MGR	12008	(null)	101	100
10	109	Daniel	Faviet	DFAVIET	515.124.4169	16-AUG-02	FI_ACCOUNT	9000	(null)	108	100
11	110	John	Chen	JCHEN	515.124.4269	28-SEP-05	FI_ACCOUNT	8200	(null)	108	100
12	111	Ismael	Sciarra	ISCIARRA	515.124.4369	30-SEP-05	FI_ACCOUNT	7700	(null)	108	100
13	112	Jose Manuel	Urman	JMURMAN	515.124.4469	07-MAR-06	FI_ACCOUNT	7800	(null)	108	100
14	113	Luis	Popp	LPOPP	515.124.4567	07-DEC-07	FI_ACCOUNT	6900	(null)	108	100
15	114	Den	Raphaely	DRAPHEAL	515.127.4561	07-DEC-02	PU_MAN	11000	(null)	100	30
16	115	Alexander	Khoo	AKHOO	515.127.4562	18-MAY-03	PU_CLERK	3100	(null)	114	30
17	116	Shelli	Baida	SBAIDA	515.127.4563	24-DEC-05	PU_CLERK	2900	(null)	114	30
18	117	Sigal	Tobias	STOBIAS	515.127.4564	24-JUL-05	PU_CLERK	2800	(null)	114	30
19	118	Guy	Himuro	GHIMURO	515.127.4565	15-NOV-06	PU_CLERK	2600	(null)	114	30
20	119	Karen	Colmenares	KCOLMENA	515.127.4566	10-AUG-07	PU_CLERK	2500	(null)	114	30
21	120	Matthew	Weiss	MWEISS	650.123.1234	18-JUL-04	ST_MAN	8000	(null)	100	50
22	121	Adam	Fripp	AFRIPP	650.123.2234	10-APR-05	ST_MAN	8200	(null)	100	50
23	122	Payam	Kaufling	PKAUFLIN	650.123.3234	01-MAY-03	ST_MAN	7900	(null)	100	50
24	123	Shanta	Vollman	SVOLLMAN	650.123.4234	10-OCT-05	ST_MAN	6500	(null)	100	50
25	124	Kevin	Mourgos	KMOURGOS	650.123.5234	16-NOV-07	ST_MAN	5800	(null)	100	50
26	125	Julia	Nayer	JNAYER	650.124.1214	16-JUL-05	ST_CLERK	3200	(null)	120	50
27	126	Irene	Mikkilineni	IMIKKILI	650.124.1224	28-SEP-06	ST_CLERK	2700	(null)	120	50
28	127	James	Landry	JLANDRY	650.124.1334	14-JAN-07	ST_CLERK	2400	(null)	120	50
29	128	Steven	Markle	SMARKLE	650.124.1434	08-MAR-08	ST_CLERK	2200	(null)	120	50
30	129	Laura	Bissot	LBISSOT	650.124.5234	20-AUG-05	ST_CLERK	3300	(null)	121	50
31	130	Mozhe	Atkinson	MATKINSO	650.124.6234	30-OCT-05	ST_CLERK	2800	(null)	121	50
32	131	James	Marlow	JAMRLOW	650.124.7234	16-FEB-05	ST_CLERK	2500	(null)	121	50
33	132	TJ	Olson	TJOLSON	650.124.8234	10-APR-07	ST_CLERK	2100	(null)	121	50
34	133	Jason	Mallin	JMALLIN	650.127.1934	14-JUN-04	ST_CLERK	3300	(null)	122	50

3. Выведите имя, фамилию, email, телефон сотрудников, которых зовут Peter.

Запрос: SELECT first_name, last_name, email, phone_number FROM HR.employees WHERE first_name = 'Peter';

<

4. Выведите имя, фамилию и идентификатор сотрудников, у которых зарплата более \$6000.

Запрос: `SELECT first_name, last_name, employee_id FROM HR.employees WHERE salary > 6000;`

Worksheet Query Builder

```
SELECT first_name, last_name, employee_id
FROM HR.employees
WHERE salary > 6000;
```

Query Result x

SQL | All Rows Fetched: 55 in 0.007 seconds

	FIRST_NAME	LAST_NAME	EMPLOYEE_ID
1	Steven	King	100
2	Neena	Kochhar	101
3	Lex	De Haan	102
4	Alexander	Hunold	103
5	Nancy	Greenberg	108
6	Daniel	Faviet	109
7	John	Chen	110
8	Ismael	Sciarra	111
9	Jose Manuel	Urman	112
10	Luis	Popp	113
11	Den	Raphaely	114
12	Matthew	Weiss	120
13	Adam	Fripp	121
14	Payam	Kaufling	122
15	Shanta	Vollman	123
16	John	Russell	145
17	Karen	Partners	146
18	Alberto	Errazuriz	147
19	Gerald	Cambrault	148
20	Eleni	Zlotkey	149
21	Peter	Tucker	150
22	David	Bernstein	151
23	Peter	Hall	152
24	Christopher	Olsen	153
25	Nanette	Cambrault	154
26	Oliver	Tuvault	155
27	Janette	King	156
28	Patrick	Sully	157
29	Allan	McEwen	158
30	Lindsey	Smith	159
31	Louise	Doran	160
32	Sarath	Sewall	161
33	Clara	Vishney	162
34	Danielle	Greene	163

5. Выведите имя, фамилию и зарплату сотрудников, которые получают больше, чем \$2500 и меньше, чем \$4000.

Запрос: SELECT first_name, last_name, salary FROM HR.employees WHERE salary > 2500 AND salary < 4000;

Worksheet

Query Builder

```
SELECT first_name, last_name, salary
FROM HR.employees
WHERE salary > 2500 AND salary < 4000;
```

Query Result x

All Rows Fetched: 31 in 0.006 seconds

	FIRST_NAME	LAST_NAME	SALARY
1	Alexander	Khoo	3100
2	Shelli	Baida	2900
3	Sigal	Tobias	2800
4	Guy	Himuro	2600
5	Julia	Nayer	3200
6	Irene	Mikkilineni	2700
7	Laura	Bissot	3300
8	Mozhe	Atkinson	2800
9	Jason	Mallin	3300
10	Michael	Rogers	2900
11	Renske	Ladwig	3600
12	Stephen	Stiles	3200
13	John	Seo	2700
14	Trenna	Rajs	3500
15	Curtis	Davies	3100
16	Randall	Matos	2600
17	Winston	Taylor	3200
18	Jean	Fleaur	3100
19	Girard	Geoni	2800
20	Julia	Dellinger	3400
21	Anthony	Cabrio	3000
22	Kelly	Chung	3800
23	Jennifer	Dilly	3600
24	Timothy	Gates	2900
25	Britney	Everett	3900
26	Samuel	McCain	3200
27	Vance	Jones	2800
28	Alana	Walsh	3100
29	Kevin	Feeney	3000
30	Donald	OConnell	2600
31	Douglas	Grant	2600

6. Выведите имя, фамилию, зарплату в долларах и зарплату в рублях (можно выбрать валюту своей страны) по текущему курсу для тех сотрудников, которые получают более 5000 рублей (считайте, что в таблице зарплата указана в долларах)

Запрос: SELECT first_name, last_name, salary AS salary_in_dollars, salary * 80.59 AS salary_in_rubles
FROM HR.employees
WHERE salary > 5000 / 80.59;

Worksheet Query Builder

```
SELECT first_name, last_name, salary AS salary_in_dollars, salary * 80.59 AS salary_in_rubles
FROM HR.employees
WHERE salary > 5000 / 80.59;
```

Query Result x

All Rows Fetched: 107 in 0.017 seconds

	FIRST_NAME	LAST_NAME	SALARY_IN_DOLLARS	SALARY_IN_RUBLES
1	Steven	King	24000	1934160
2	Neena	Kochhar	17000	1370030
3	Lex	De Haan	17000	1370030
4	Alexander	Hunold	9000	725310
5	Bruce	Ernst	6000	483540
6	David	Austin	4800	386832
7	Valli	Pataballa	4800	386832
8	Diana	Lorentz	4200	338478
9	Nancy	Greenberg	12008	967724.72
10	Daniel	Faviet	9000	725310
11	John	Chen	8200	660838
12	Ismael	Sciarra	7700	620543
13	Jose Manuel	Urman	7800	628602
14	Luis	Popp	6900	556071
15	Den	Raphaely	11000	886490
16	Alexander	Khoo	3100	249829
17	Shelli	Baida	2900	233711
18	Sigal	Tobias	2800	225652
19	Guy	Himuro	2600	209534
20	Karen	Colmenares	2500	201475
21	Matthew	Weiss	8000	644720
22	Adam	Fripp	8200	660838
23	Payam	Kaufling	7900	636661
24	Shanta	Vollman	6500	523835
25	Kevin	Mourgos	5800	467422
26	Julia	Nayer	3200	257888
27	Irene	Mikkilineni	2700	217593
28	James	Landry	2400	193416
29	Steven	Markle	2200	177298
30	Laura	Bissot	3300	265947
31	Mozhe	Atkinson	2800	225652
32	James	Marlow	2500	201475
33	TJ	Olson	2100	169239
34	Jason	Mallin	3300	265947

7. Выведите все адреса из Великобритании с указанием отделов, которые по этим адресам находятся. (связь таблиц Countries, Locations).

```
FROM HR.employees
  JOIN HR.departments USING (department_id)
  JOIN HR.locations USING (location_id)
  JOIN HR.countries USING (country_id)
WHERE country_name = 'United Kingdom';
```

The screenshot shows a database query tool interface. The top part displays a SQL query in the 'Query Builder' tab. The query is:
`SELECT DISTINCT street_address, department_name
FROM HR.employees
 JOIN HR.departments USING (department_id)
 JOIN HR.locations USING (location_id)
 JOIN HR.countries USING (country_id)
WHERE country_name = 'United Kingdom';`
The bottom part shows the 'Query Result' tab with the following data:

STREET_ADDRESS	DEPARTMENT_NAME
1 8204 Arthur St	Human Resources
2 Magdalen Centre, The Oxford Science Park	Sales

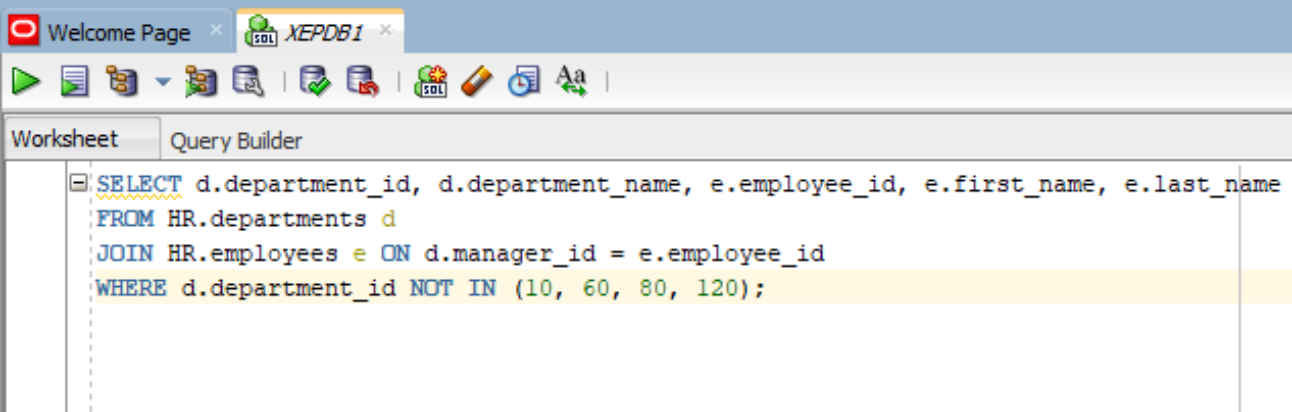
8. Выведите все страны, название которых содержит букву "i". Вывести страну и город.

Запрос: SELECT DISTINCT country_name, city
FROM HR.countries
JOIN HR.countries USING (country_id)
WHERE country_name LIKE '%i%';

Worksheet		Query Builder	
		<pre>SELECT DISTINCT country_name, city FROM HR.locations JOIN HR.countries USING (country_id) WHERE country_name LIKE '%i%';</pre>	
Script Output x		Query Result x	
		SQL All Rows Fetched: 15 in 0.007 seconds	
COUNTRY_NAME		CITY	
1	Australia	Sydney	
2	Brazil	Sao Paulo	
3	Switzerland	Geneva	
4	Switzerland	Bern	
5	China	Beijing	
6	India	Bombay	
7	Mexico	Mexico City	
8	Singapore	Singapore	
9	United Kingdom	London	
10	United Kingdom	Oxford	
11	United Kingdom	Stretford	
12	United States of America	Southlake	
13	United States of America	South San Francisco	
14	United States of America	South Brunswick	
15	United States of America	Seattle	

9. Выведите менеджеров для всех отделов кроме 10, 60, 80, 120.

Запрос: SELECT d.department_id, d.department_name, e.employee_id, e.first_name, e.last_name
FROM HR.departments d
JOIN HR.employees e ON d.manager_id = e.employee_id
WHERE d.department_id NOT IN (10, 60, 80, 120);



The screenshot shows the 'Query Result' tab with the query results. The results are displayed in a table with 6 columns: DEPARTMENT_ID, DEPARTMENT_NAME, EMPLOYEE_ID, FIRST_NAME, and LAST_NAME. The table contains 8 rows of data.

	DEPARTMENT_ID	DEPARTMENT_NAME	EMPLOYEE_ID	FIRST_NAME	LAST_NAME
1	90	Executive	100	Steven	King
2	100	Finance	108	Nancy	Greenberg
3	30	Purchasing	114	Den	Raphaely
4	50	Shipping	121	Adam	Fripp
5	20	Marketing	201	Michael	Hartstein
6	40	Human Resources	203	Susan	Mavris
7	70	Public Relations	204	Hermann	Baer
8	110	Accounting	205	Shelley	Higgins