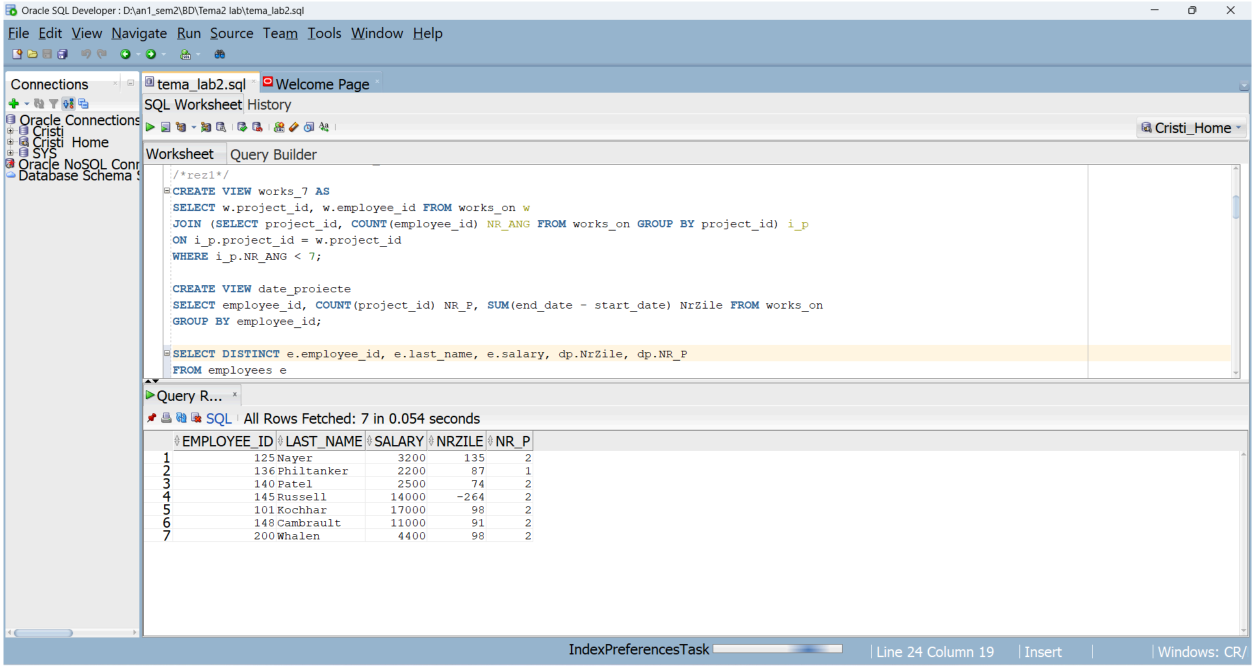
```sql
/*
1. (8p) Pentru fiecare proiect in cadrul caruia numarul de angajati care au
lucrat la acesta este mai mic decat 7, sa se afiseze angajatii care au lucrat
in toate aceste proiecte. Se vor afisa id-ul angajatului, numele acestuia,
salariul, numarul de zile lucrate in cadrul respectivelor proiecte (numarul
de zile se calculeaza in functie de -> start_date – data la care a inceput
lucrul, end_date – data la care a finalizat lucrul) – coloana o sa se
numeasca NrZile. De asemenea, sa se afiseze si o coloana numita
Nr. total proiecte – aceasta o sa contina numarul total de proiecte lucrate
de angajatul respectiv.
*/
/*OBS: angajatul 145 are end_date inainte de start deci si nr de zile va fii pe minus*/
/*rez1*/
CREATE VIEW works_7 AS
SELECT w.project_id, w.employee_id FROM works_on w
JOIN (SELECT project_id, COUNT(employee_id) NR_ANG FROM works_on GROUP BY
project_id) i_p
ON i_p.project_id = w.project_id
WHERE i_p.NR_ANG < 7;

CREATE VIEW date_proiecte
SELECT employee_id, COUNT(project_id) NR_P, SUM(end_date - start_date) NrZile FROM
works_on
GROUP BY employee_id;

SELECT DISTINCT e.employee_id, e.last_name, e.salary, dp.NrZile, dp.NR_P
FROM employees e
JOIN works_7 w ON w.employee_id = e.employee_id
JOIN date_proiecte dp ON dp.employee_id = e.employee_id;
```

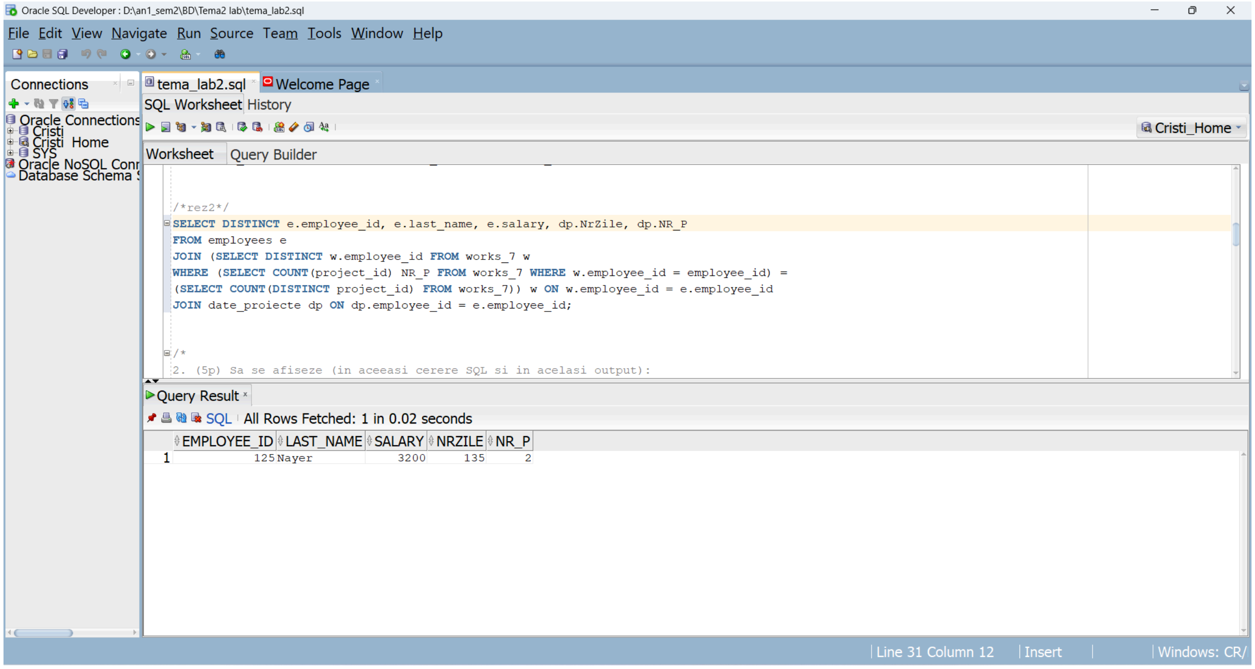


```
/*rez1*/
CREATE VIEW works_7 AS
SELECT w.project_id, w.employee_id FROM works_on w
JOIN (SELECT project_id, COUNT(employee_id) NR_ANG FROM works_on GROUP BY project_id) i_p
ON i_p.project_id = w.project_id
WHERE i_p.NR_ANG < 7;

CREATE VIEW date_proiecte
SELECT employee_id, COUNT(project_id) NR_P, SUM(end_date - start_date) NrZile FROM works_on
GROUP BY employee_id;

SELECT DISTINCT e.employee_id, e.last_name, e.salary, dp.NrZile, dp.NR_P
FROM employees e
```

All Rows Fetched: 7 in 0.054 seconds

	EMPLOYEE_ID	LAST_NAME	SALARY	NRZILE	NR_P
1	125	Nayer	3200	135	2
2	136	Philtanker	2200	87	1
3	140	Patel	2500	74	2
4	145	Russell	14000	-264	2
5	101	Kochhar	17000	98	2
6	148	Cambrault	11000	91	2
7	200	Whalen	4400	98	2

```
/*rez2*/
SELECT DISTINCT e.employee_id, e.last_name, e.salary, dp.NrZile, dp.NR_P
FROM employees e
JOIN (SELECT DISTINCT w.employee_id FROM works_7 w
WHERE (SELECT COUNT(project_id) NR_P FROM works_7 WHERE w.employee_id =
employee_id) =
(SELECT COUNT(DISTINCT project_id) FROM works_7)) w ON w.employee_id =
e.employee_id
JOIN date_proiecte dp ON dp.employee_id = e.employee_id;
```



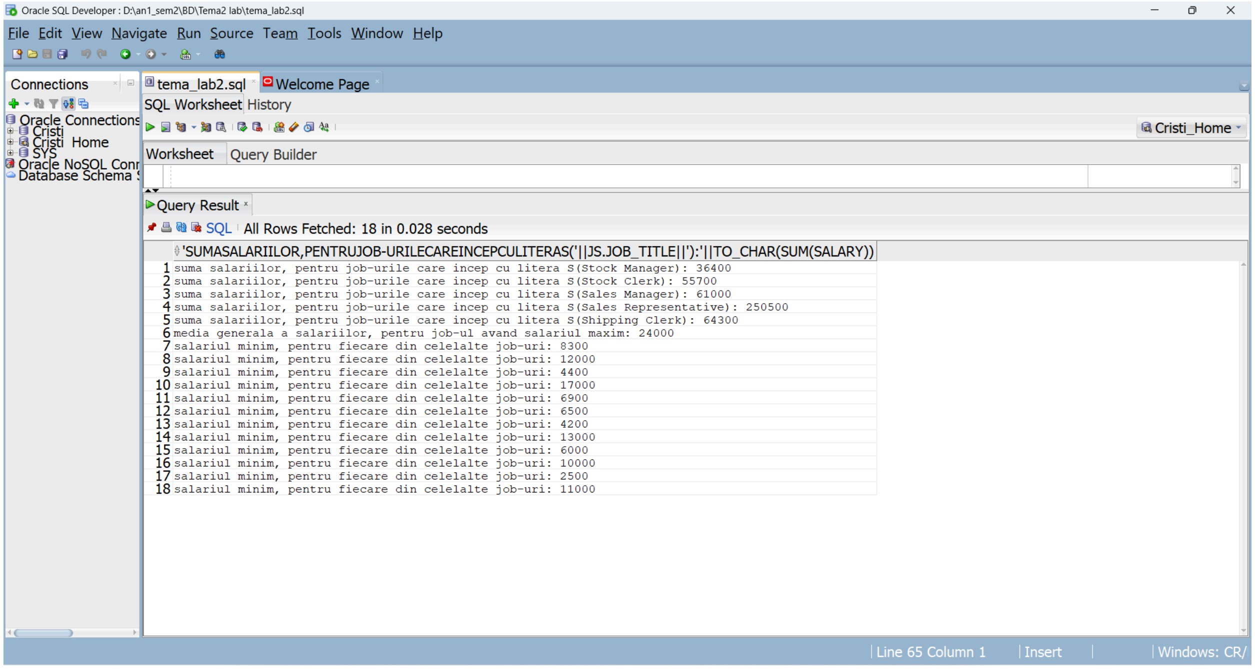
```
/*rez2*/
SELECT DISTINCT e.employee_id, e.last_name, e.salary, dp.NrZile, dp.NR_P
FROM employees e
JOIN (SELECT DISTINCT w.employee_id FROM works_7 w
WHERE (SELECT COUNT(project_id) NR_P FROM works_7 WHERE w.employee_id = employee_id) =
(SELECT COUNT(DISTINCT project_id) FROM works_7)) w ON w.employee_id = e.employee_id
JOIN date_proiecte dp ON dp.employee_id = e.employee_id;


/*
2. (5p) Sa se afiseze (in aceeasi cerere SQL si in acelasi output):
```

All Rows Fetched: 1 in 0.02 seconds

	EMPLOYEE_ID	LAST_NAME	SALARY	NRZILE	NR_P
1	125	Nayer	3200	135	2

```
/*
2. (5p) Sa se afiseze (in aceeasi cerere SQL si in acelasi output):
a. suma salariilor, pentru job-urile care incep cu litera S;
b. media generala a salariilor, pentru job-ul avand salariul maxim;
c. salariul minim, pentru fiecare din celelalte job-uri.
*/

/*view pt cerinta a)*/
CREATE VIEW joburi_S AS
   SELECT * FROM jobs WHERE UPPER(job_title) LIKE 'S%';
/*view pt cerinta b)*/
CREATE VIEW super_job AS
   SELECT * FROM jobs WHERE max_salary = (SELECT MAX(max_salary) FROM jobs);
/*view pt cerinta c)*/
CREATE VIEW celalalte_joburi AS
   SELECT * FROM jobs
   MINUS
   (SELECT * FROM joburi_s UNION SELECT * FROM super_job);


/*a)*/
SELECT 'suma salariilor, pentru job-urile care incep cu litera S(' || js.job_title || '): ' ||
to_char(SUM(salary)) FROM employees e
JOIN joburi_s js ON js.job_id = e.job_id
GROUP BY js.job_id, js.job_title
```

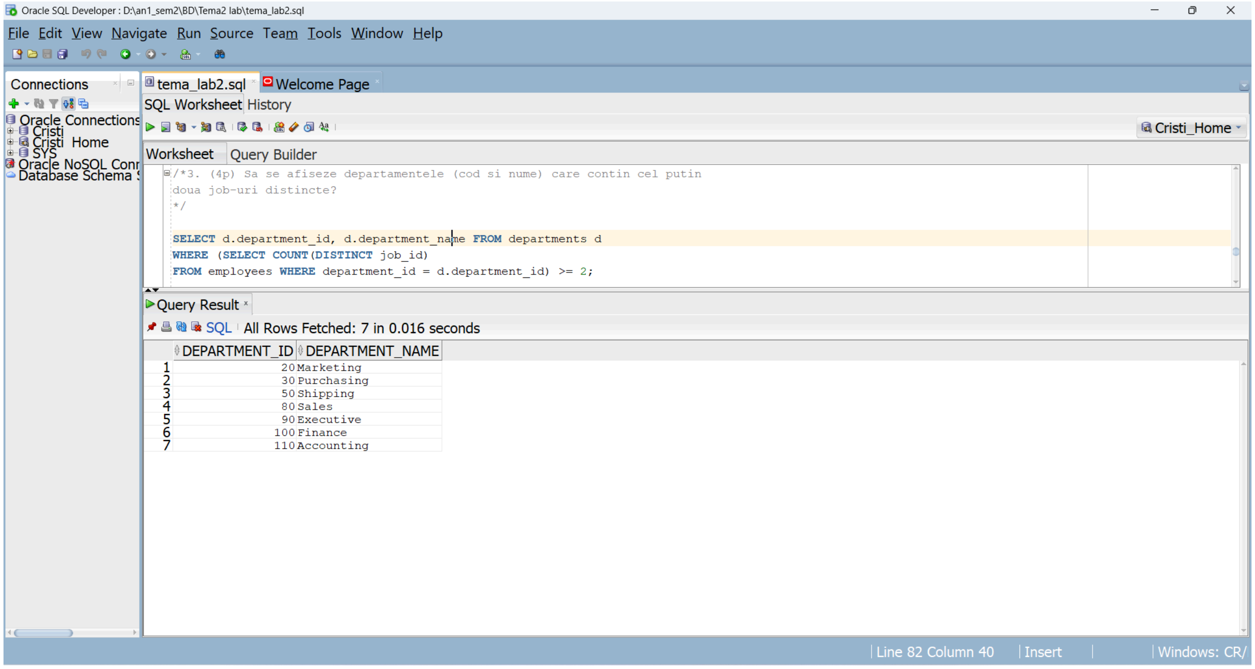
```
UNION

/*b)*/
SELECT 'media generala a salariilor, pentru job-ul avand salariul maxim: ' ||
to_char(AVG(e.salary)) FROM employees e
JOIN super_job sj ON sj.job_id = e.job_id
GROUP BY sj.job_id

UNION
/*c)*/
SELECT 'salariul minim, pentru fiecare din celelalte job-uri: ' || to_char(MIN(salary)) FROM
employees e
JOIN celelalte_joburi cj ON e.job_id = cj.job_id
GROUP BY cj.job_id;
```



All Rows Fetched: 18 in 0.028 seconds

	'SUMASALARIILOR,PENTRUJOB-URILECAREINCEPCULITERAS('||JS.JOB_TITLE||'):'||TO_CHAR(SUM(SALARY))
1	suma salariilor, pentru job-urile care incep cu litera S(Stock Manager): 36400
2	suma salariilor, pentru job-urile care incep cu litera S(Stock Clerk): 55700
3	suma salariilor, pentru job-urile care incep cu litera S(Sales Manager): 61000
4	suma salariilor, pentru job-urile care incep cu litera S(Sales Representative): 250500
5	suma salariilor, pentru job-urile care incep cu litera S(Shipping Clerk): 64300
6	media generala a salariilor, pentru job-ul avand salariul maxim: 24000
7	salariul minim, pentru fiecare din celelalte job-uri: 8300
8	salariul minim, pentru fiecare din celelalte job-uri: 12000
9	salariul minim, pentru fiecare din celelalte job-uri: 4400
10	salariul minim, pentru fiecare din celelalte job-uri: 17000
11	salariul minim, pentru fiecare din celelalte job-uri: 6900
12	salariul minim, pentru fiecare din celelalte job-uri: 6500
13	salariul minim, pentru fiecare din celelalte job-uri: 4200
14	salariul minim, pentru fiecare din celelalte job-uri: 13000
15	salariul minim, pentru fiecare din celelalte job-uri: 6000
16	salariul minim, pentru fiecare din celelalte job-uri: 10000
17	salariul minim, pentru fiecare din celelalte job-uri: 2500
18	salariul minim, pentru fiecare din celelalte job-uri: 11000

```
/*3. (4p) Sa se afiseze departamentele (cod si nume) care contin cel putin
doua job-uri distincte?
*/

SELECT d.department_id, d.department_name FROM departments d
WHERE (SELECT COUNT(DISTINCT job_id)
FROM employees WHERE department_id = d.department_id) >= 2;
```

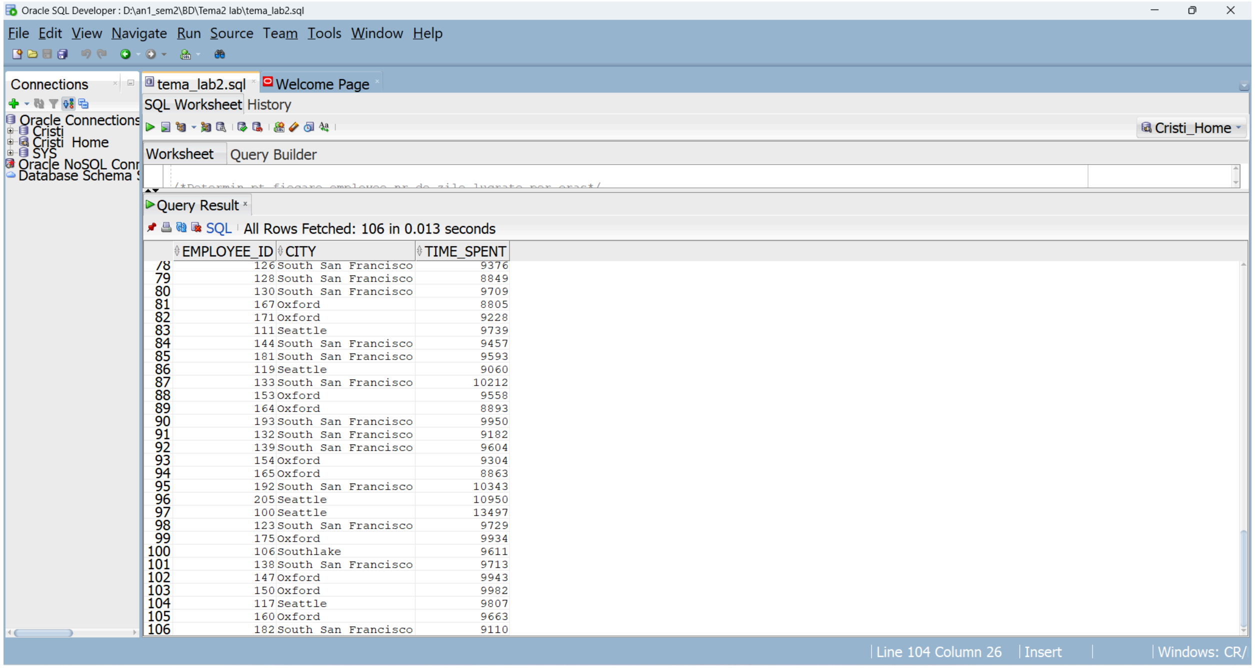


```
/*3. (4p) Sa se afiseze departamentele (cod si nume) care contin cel putin
doua job-uri distincte?
*/

SELECT d.department_id, d.department_name FROM departments d
WHERE (SELECT COUNT(DISTINCT job_id)
FROM employees WHERE department_id = d.department_id) >= 2;
```

All Rows Fetched: 7 in 0.016 seconds

```
   DEPARTMENT_ID DEPARTMENT_NAME
1             20 Marketing
2             30 Purchasing
3             50 Shipping
4             80 Sales
5             90 Executive
6            100 Finance
7            110 Accounting
```

```
/*
4. (8p) Sa se listeze pentru fiecare angajat orasul in care a lucrat cele mai
multe zile (VEZI exemplul de mai jos).
*/

/*Determin pt fiecare employee nr de zile lucrate per oras*/
CREATE VIEW emp_oras AS
SELECT e.employee_id, l.city,
nvl(SUM(j.end_date - j.start_date) + TRUNC(SYSDATE - e.hire_date),
TRUNC(SYSDATE - e.hire_date)) TIME_SPENT
FROM locations l
JOIN departments d ON d.location_id = l.location_id
JOIN employees e ON e.department_id = d.department_id
LEFT JOIN job_history j ON e.employee_id = j.employee_id
GROUP BY l.city, e.employee_id, e.hire_date;

/*Aleg orasul in care a lucrat cel mai mult*/
SELECT eo.employee_id, eo.city, eo.time_spent FROM emp_oras eo
WHERE eo.time_spent =
(SELECT MAX(TIME_SPENT) FROM emp_oras
WHERE eo.employee_id = employee_id);
```



```
/*Determin pt fiecare employee nr de zile lucrate per oras*/
```

All Rows Fetched: 106 in 0.013 seconds

	EMPLOYEE_ID	CITY	TIME_SPENT
78	126	South San Francisco	9376
79	128	South San Francisco	8849
80	130	South San Francisco	9709
81	167	Oxford	8805
82	171	Oxford	9228
83	111	Seattle	9739
84	144	South San Francisco	9457
85	181	South San Francisco	9593
86	119	Seattle	9060
87	133	South San Francisco	10212
88	153	Oxford	9558
89	164	Oxford	8893
90	193	South San Francisco	9950
91	132	South San Francisco	9182
92	139	South San Francisco	9604
93	154	Oxford	9304
94	165	Oxford	8863
95	192	South San Francisco	10343
96	205	Seattle	10950
97	100	Seattle	13497
98	123	South San Francisco	9729
99	175	Oxford	9934
100	106	Southlake	9611
101	138	South San Francisco	9713
102	147	Oxford	9943
103	150	Oxford	9982
104	117	Seattle	9807
105	160	Oxford	9663
106	182	South San Francisco	9110