DISTRIBUTED AND PARALLEL DATABASES

PART-4

The library management system manages the catalog of a library as well as save the time of people. This system helps to keep the records of whole transactions of the books available in the library, gives the availability of a specific book along with it specifying its placement in the department. This system allows people to reserve cabin for a particular time. It also keeps track of the donations and funds received. This system stores details of the employees, specific details of books and users. This management system is available in many cities. So, the services provided are availed by libraries in all cities.

To build this project we use Oracle Enterprise 21c Edition Database. This can be used as a local database storage and in future can also be used to connect to distributed databases across the globe. This system can also be used for creating database links (public and private), that can be used to connect different libraries in the project and access data across them.

Here we have 7 entities. They are Employees, Books, Users, Cabins, Departments, funds and Donations, Library.

- **Libraries** are available in different places and each one has its own name. They have been identified by a unique id, Library_ID.
- **Employees** are specific to a single work location. They have a ID which is specific to a particular employee. Therefore, no two employees have the same EMP ID. This entity shares a many to one relationship with libraries.
- **Books** details are stored in this table. The primary key for this table is Book_ID. Libraries in various locations will have copies of a same book. So here it shares a many to many relationships with library.
- **Cabin** stores info about various bookings made and the availability status of the pods. Here a one-to-many relation is seen where one library will have many cabins in it.
- **Department** has its own Department_ID. Each department has more than one copy of a single book, hence showing a one-to-many relationship.
- **Funds and Donations** has info about the funds received from governments and any other sources.
- **Users** have details about each user. One user can borrow more than one book but are restricted to borrow more than five books at a time.

2)Transformation of E-R diagram to Database Schema:

- Library (l_id, name address)
- Employees (e_id, name, salary, contact, role, doj, l_id)
- Cabin (cabin_id, availability, l_id)
- Books (b_id, ISBN, name, author, status, l_id, dept_id)
- Books_transactions (b_id, u_id, t_id, t_date)
- Departments(dept_id, dept_name, instore_location)
- Users (u_id, u_name, address, contact, dob, gender)
- User_checkin (u_id, l_id, checkin_id, checkin_date)
- Donations (d_id, d_date, amount, u_id)

Here in this part, we are expected to create a distributed database system. To access each fragment in individual sites we have to use a global DB and execute our applications. For this we are creating a link for the global database and accessing it.

The below screenshot shows the link creation.

To perform these tasks by accessing global database using database link.

Next part we are creating fragments for out database which fulfills our aim of creating a distributed database. We now fragment our database tables based on the location . So each table that we work on is now fragmented based on a particular location . Each site represents one location and each person in the team is working on one location each.

Here we fragmented out Database in the following manner:

The global database is present with Sreekar who represent site India . Other sites access the database from sreekar(Site India)

The other locations that we are working with are as follows Canada,France,England , USA.

Site India: Sreekar

Site Canada: Lavanya

Site France: Sharanya

Site England: Manish

Site USA: Vishnu

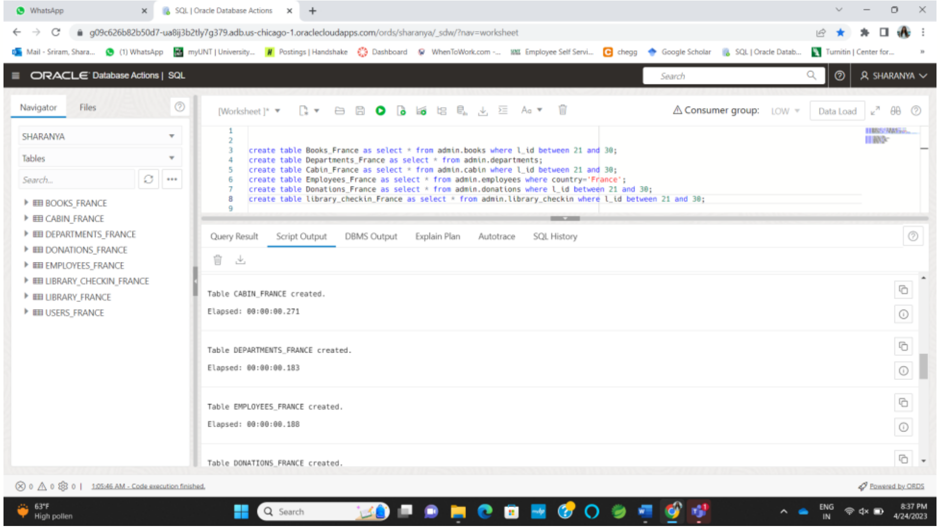
The global location database accessed by site India gives/grants permissions to the rest of them using the following command .

```
grant select on users_india to sharanya;
grant select on users_india to vishnu;
grant select on users_india to lavanya;
grant select on users_india to manish;
```

This way it works for all users but varies with table name .

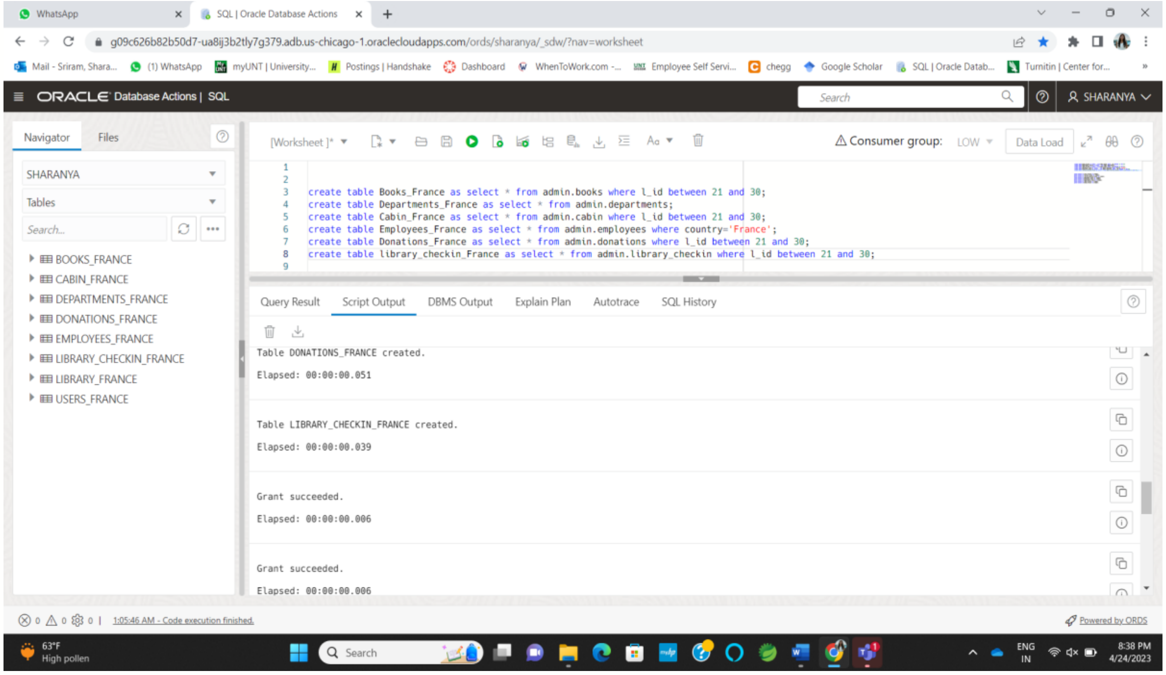
We create fragments of tables based on locations using the following commands .
Here lets take an example of Site France .

```
create table Books_France as select * from admin.books where l_id between 21 and 30;

create table Departments_France as select * from admin.departments;

create table Cabin_France as select * from admin.cabin where l_id between 21 and 30;

create table Employees_France as select * from admin.employees where country='France';

create table Donations_France as select * from admin.donations where l_id between 21 and 30;

create table library_checkin_France as select * from admin.library_checkin where l_id between 21 and 30;
```



```
create table Books_France as select * from admin.books where l_id between 21 and 30;
create table Departments_France as select * from admin.departments;
create table Cabin_France as select * from admin.cabin where l_id between 21 and 30;
create table Employees_France as select * from admin.employees where country='France';
create table Donations_France as select * from admin.donations where l_id between 21 and 30;
create table library_checkin_France as select * from admin.library_checkin where l_id between 21 and 30;
```

```
Table CABIN_FRANCE created.

Elapsed: 00:00:00.271

Table DEPARTMENTS_FRANCE created.

Elapsed: 00:00:00.183

Table EMPLOYEES_FRANCE created.

Elapsed: 00:00:00.188

Table DONATIONS_FRANCE created.
```



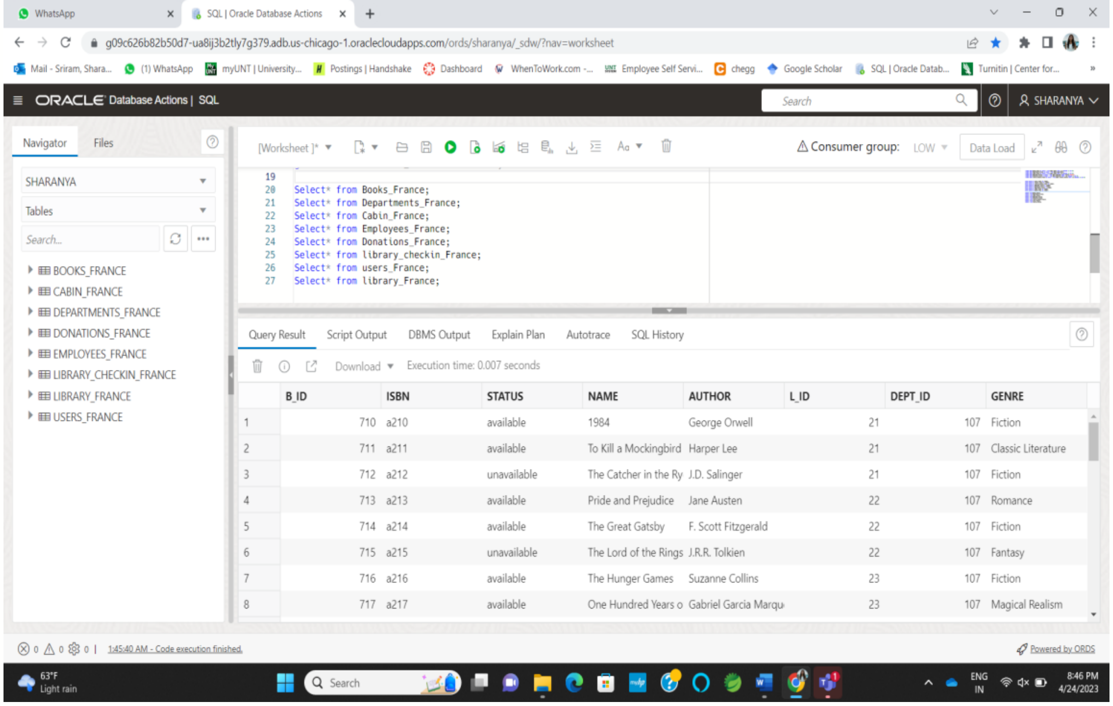
```
create table Books_France as select * from admin.books where l_id between 21 and 30;
create table Departments_France as select * from admin.departments;
create table Cabin_France as select * from admin.cabin where l_id between 21 and 30;
create table Employees_France as select * from admin.employees where country='France';
create table Donations_France as select * from admin.donations where l_id between 21 and 30;
create table library_checkin_France as select * from admin.library_checkin where l_id between 21 and 30;
```

```
Table DONATIONS_FRANCE created.

Elapsed: 00:00:00.051


Table LIBRARY_CHECKIN_FRANCE created.

Elapsed: 00:00:00.039


Grant succeeded.

Elapsed: 00:00:00.006


Grant succeeded.

Elapsed: 00:00:00.006
```

This way fragments of the database is created with every site based on location.

The data screenshots after fragmenting data from global database is as follows:

```
Select* from Books_France;
```

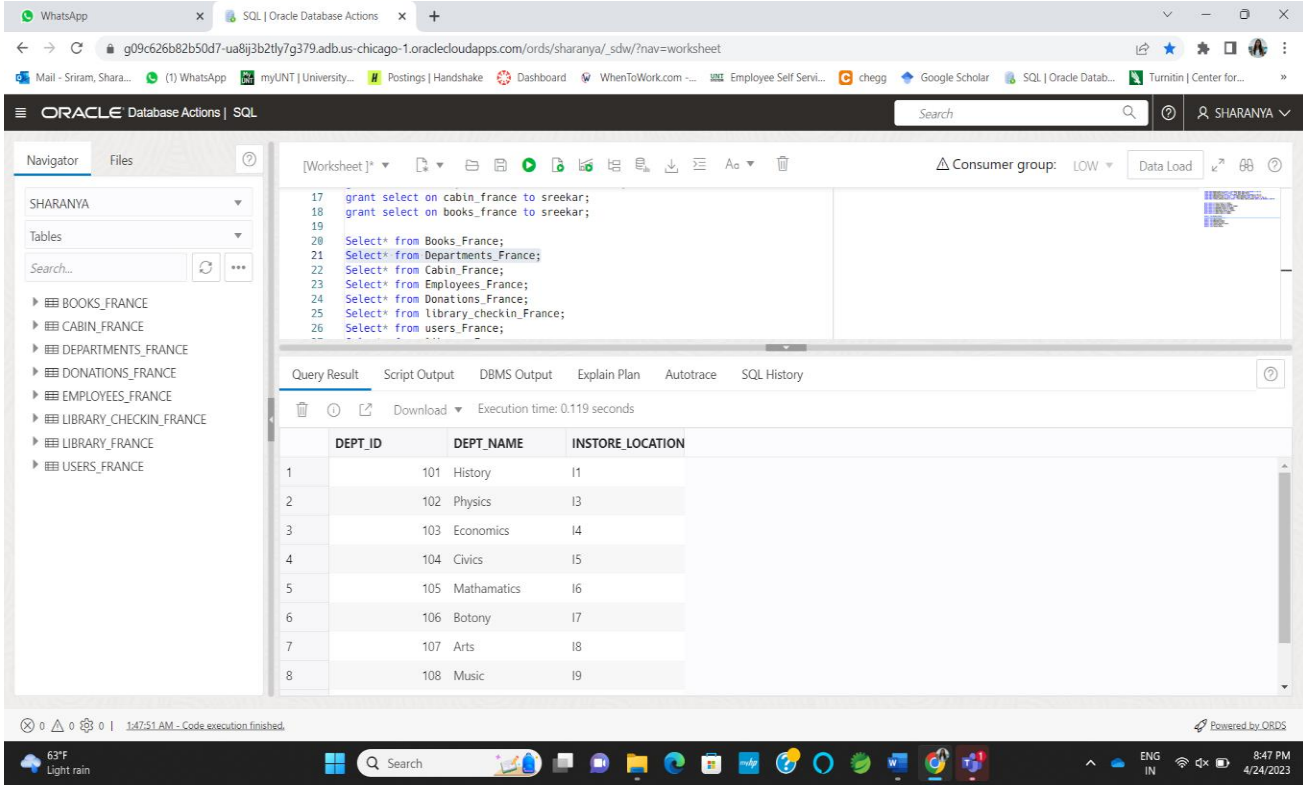


```
Select* from Books_France;
Select* from Departments_France;
Select* from Cabin_France;
Select* from Employees_France;
Select* from Donations_France;
Select* from library_checkin_France;
Select* from users_France;
Select* from library_France;
```

Execution time: 0.007 seconds

	B_ID	ISBN	STATUS	NAME	AUTHOR	L_ID	DEPT_ID	GENRE
1	710	a210	available	1984	George Orwell	21	107	Fiction
2	711	a211	available	To Kill a Mockingbird	Harper Lee	21	107	Classic Literature
3	712	a212	unavailable	The Catcher in the Ry	J.D. Salinger	21	107	Fiction
4	713	a213	available	Pride and Prejudice	Jane Austen	22	107	Romance
5	714	a214	available	The Great Gatsby	F. Scott Fitzgerald	22	107	Fiction
6	715	a215	unavailable	The Lord of the Rings	J.R.R. Tolkien	22	107	Fantasy
7	716	a216	available	The Hunger Games	Suzanne Collins	23	107	Fiction
8	717	a217	available	One Hundred Years o	Gabriel Garcia Marqu	23	107	Magical Realism

```
Select* from Departments_France;
```



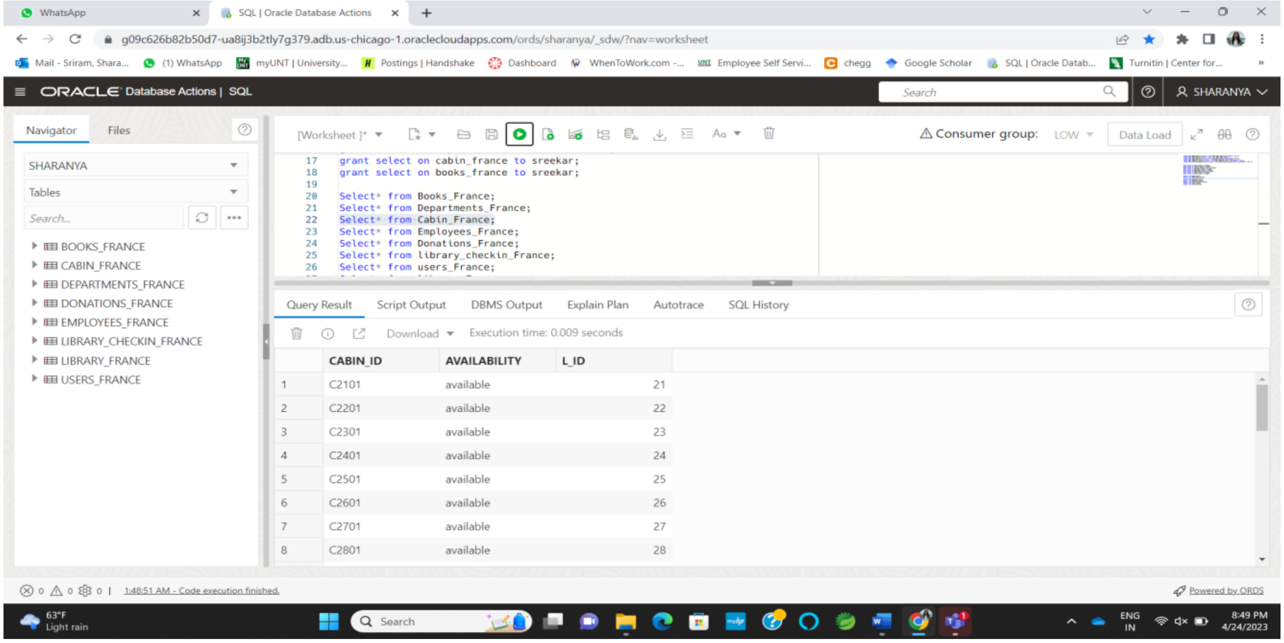
```
grant select on cabin_france to sreekar;
grant select on books_france to sreekar;

Select* from Books_France;
Select* from Departments_France;
Select* from Cabin_France;
Select* from Employees_France;
Select* from Donations_France;
Select* from library_checkin_France;
Select* from users_France;
```

Execution time: 0.119 seconds

	DEPT_ID	DEPT_NAME	INSTORE_LOCATION
1	101	History	I1
2	102	Physics	I3
3	103	Economics	I4
4	104	Civics	I5
5	105	Mathamatics	I6
6	106	Botony	I7
7	107	Arts	I8
8	108	Music	I9

1:47:51 AM - Code execution finished.

```
Select* from Departments_France;
```



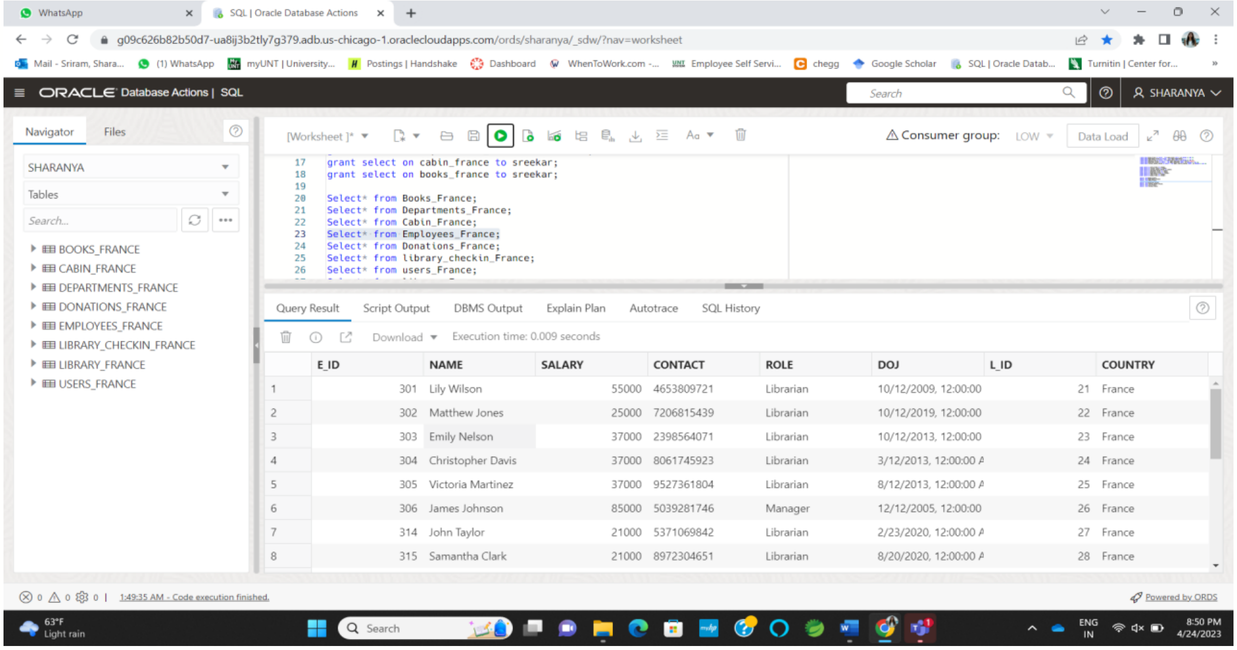
```
grant select on cabin_france to sreekar;
grant select on books_france to sreekar;

Select* from Books_France;
Select* from Departments_France;
Select* from Cabin_France;
Select* from Employees_France;
Select* from Donations_France;
Select* from library_checkin_France;
Select* from users_France;
```

Execution time: 0.009 seconds

	CABIN_ID	AVAILABILITY	L_ID
1	C2101	available	21
2	C2201	available	22
3	C2301	available	23
4	C2401	available	24
5	C2501	available	25
6	C2601	available	26
7	C2701	available	27
8	C2801	available	28

1:48:51 AM - Code execution finished.

```
Select * from Employees_France;
```



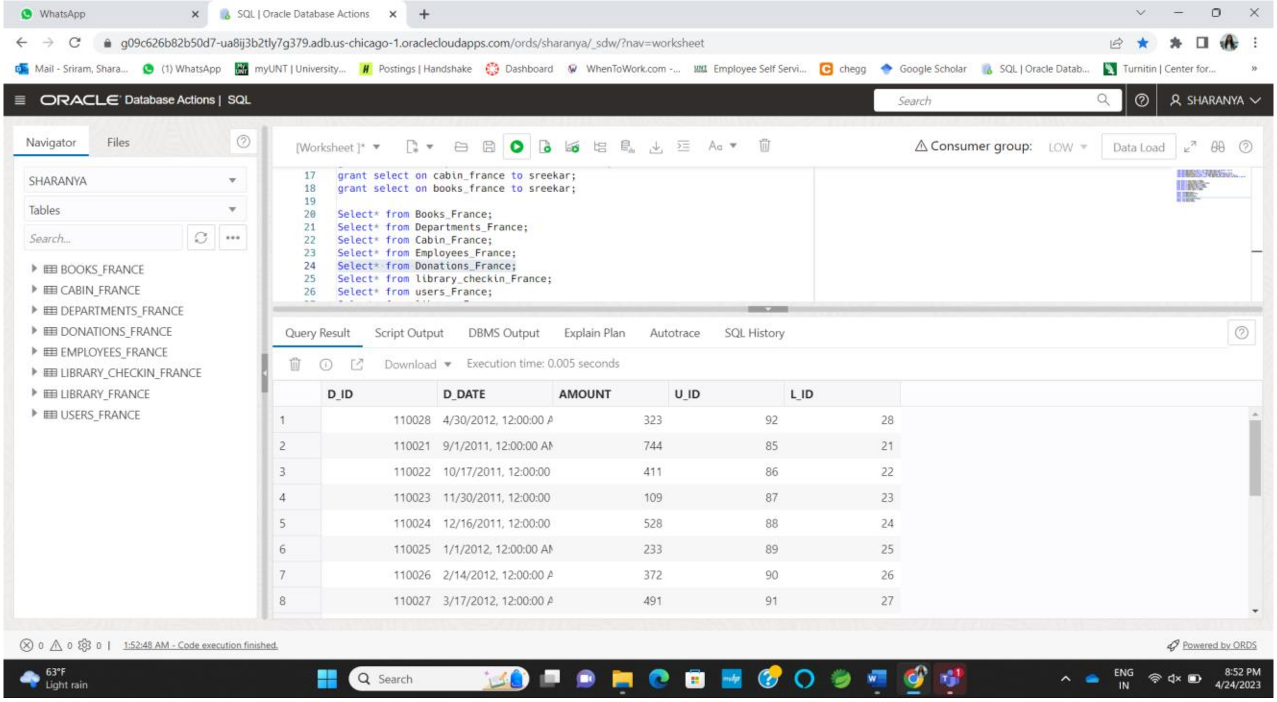
```
grant select on cabin_france to sreekar;
grant select on books_france to sreekar;

Select* from Books_France;
Select* from Departments_France;
Select* from Cabin_France;
Select* from Employees_France;
Select* from Donations_France;
Select* from library_checkin_France;
Select* from users_France;
```

Execution time: 0.009 seconds

	E_ID	NAME	SALARY	CONTACT	ROLE	DOJ	L_ID	COUNTRY
1	301	Lily Wilson	55000	4653809721	Librarian	10/12/2009, 12:00:00	21	France
2	302	Matthew Jones	25000	7206815439	Librarian	10/12/2019, 12:00:00	22	France
3	303	Emily Nelson	37000	2398564071	Librarian	10/12/2013, 12:00:00	23	France
4	304	Christopher Davis	37000	8061745923	Librarian	3/12/2013, 12:00:00 A	24	France
5	305	Victoria Martinez	37000	9527361804	Librarian	8/12/2013, 12:00:00 A	25	France
6	306	James Johnson	85000	5039281746	Manager	12/12/2005, 12:00:00	26	France
7	314	John Taylor	21000	5371069842	Librarian	2/23/2020, 12:00:00 A	27	France
8	315	Samantha Clark	21000	8972304651	Librarian	8/20/2020, 12:00:00 A	28	France

1:49:35 AM - Code execution finished.

```
Select* from Donations_France;
```



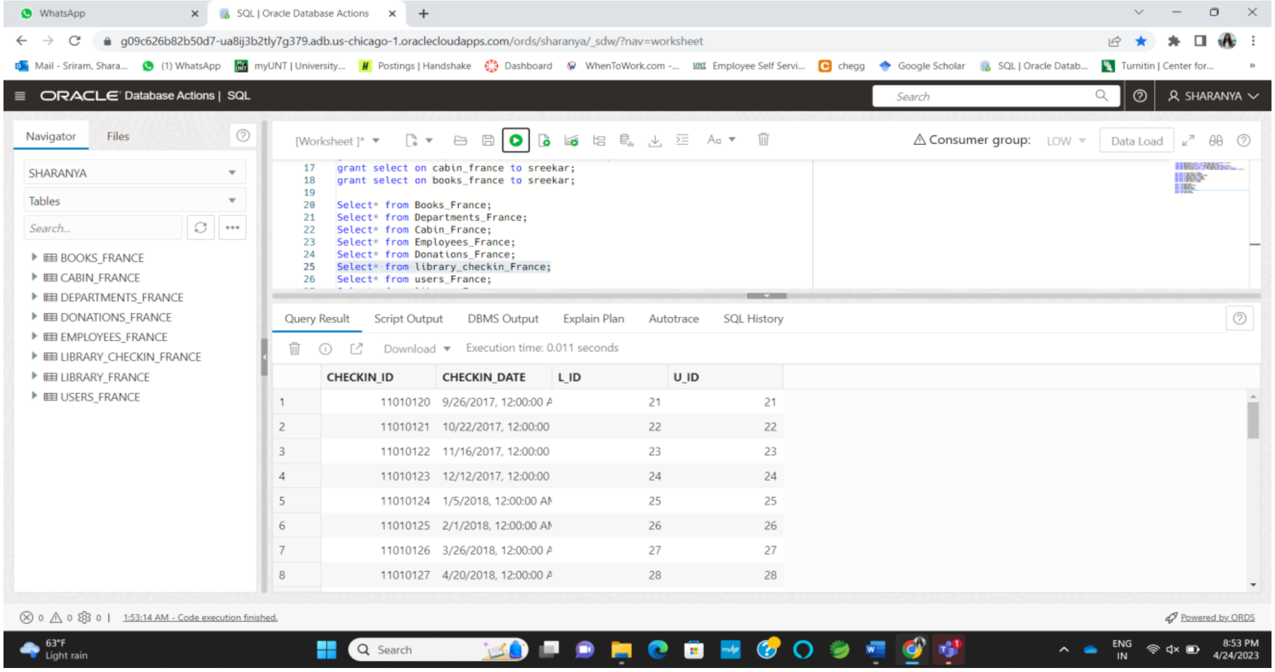
```
grant select on cabin_france to sreekar;
grant select on books_france to sreekar;

Select* from Books_France;
Select* from Departments_France;
Select* from Cabin_France;
Select* from Employees_France;
Select* from Donations_France;
Select* from library_checkin_France;
Select* from users_France;
```

Execution time: 0.005 seconds

	D_ID	D_DATE	AMOUNT	U_ID	L_ID
1	110028	4/30/2012, 12:00:00 A	323	92	28
2	110021	9/1/2011, 12:00:00 AI	744	85	21
3	110022	10/17/2011, 12:00:00	411	86	22
4	110023	11/30/2011, 12:00:00	109	87	23
5	110024	12/16/2011, 12:00:00	528	88	24
6	110025	1/1/2012, 12:00:00 AI	233	89	25
7	110026	2/14/2012, 12:00:00 A	372	90	26
8	110027	3/17/2012, 12:00:00 A	491	91	27

1:52:48 AM - Code execution finished.

```
Select* from library_checkin_France;
```



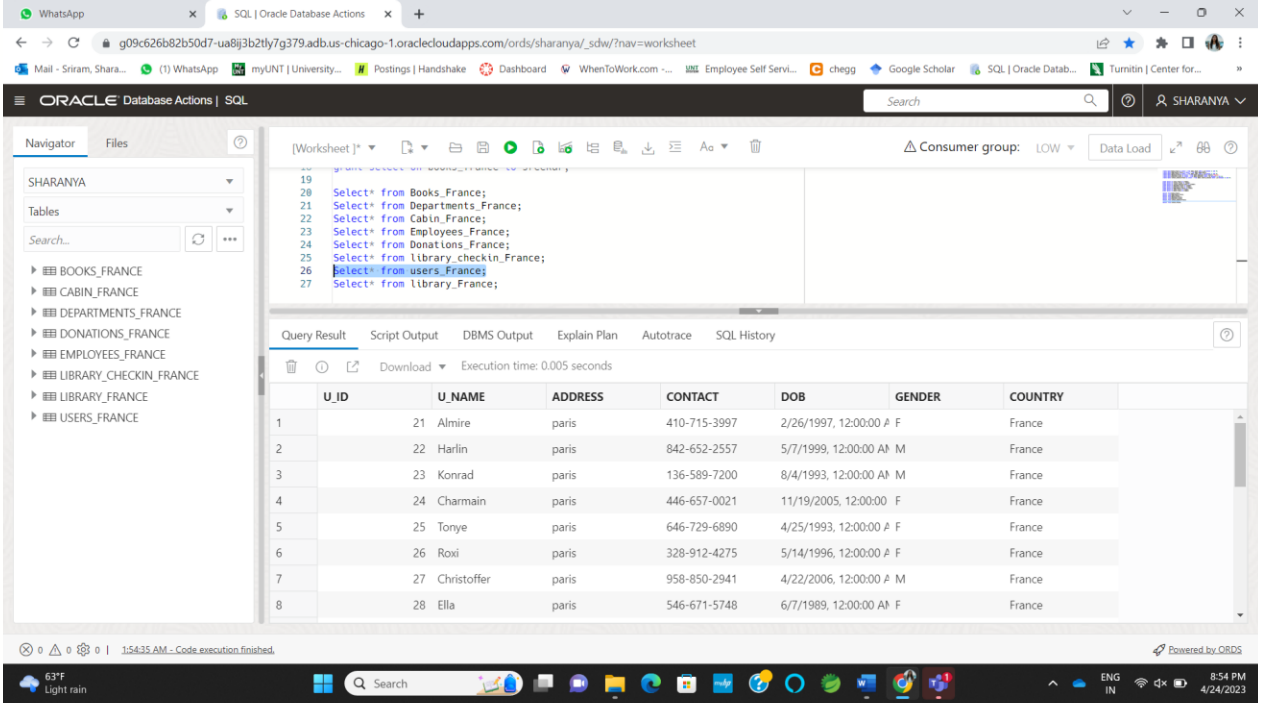
```
grant select on cabin_france to sreekar;
grant select on books_france to sreekar;

Select* from Books_France;
Select* from Departments_France;
Select* from Cabin_France;
Select* from Employees_France;
Select* from Donations_France;
Select* from library_checkin_France;
Select* from users_France;
```

Execution time: 0.011 seconds

	CHECKIN_ID	CHECKIN_DATE	L_ID	U_ID
1	11010120	9/26/2017, 12:00:00 A	21	21
2	11010121	10/22/2017, 12:00:00	22	22
3	11010122	11/16/2017, 12:00:00	23	23
4	11010123	12/12/2017, 12:00:00	24	24
5	11010124	1/5/2018, 12:00:00 AI	25	25
6	11010125	2/1/2018, 12:00:00 AI	26	26
7	11010126	3/26/2018, 12:00:00 A	27	27
8	11010127	4/20/2018, 12:00:00 A	28	28

1:53:14 AM - Code execution finished.

```
Select* from users_France;
```



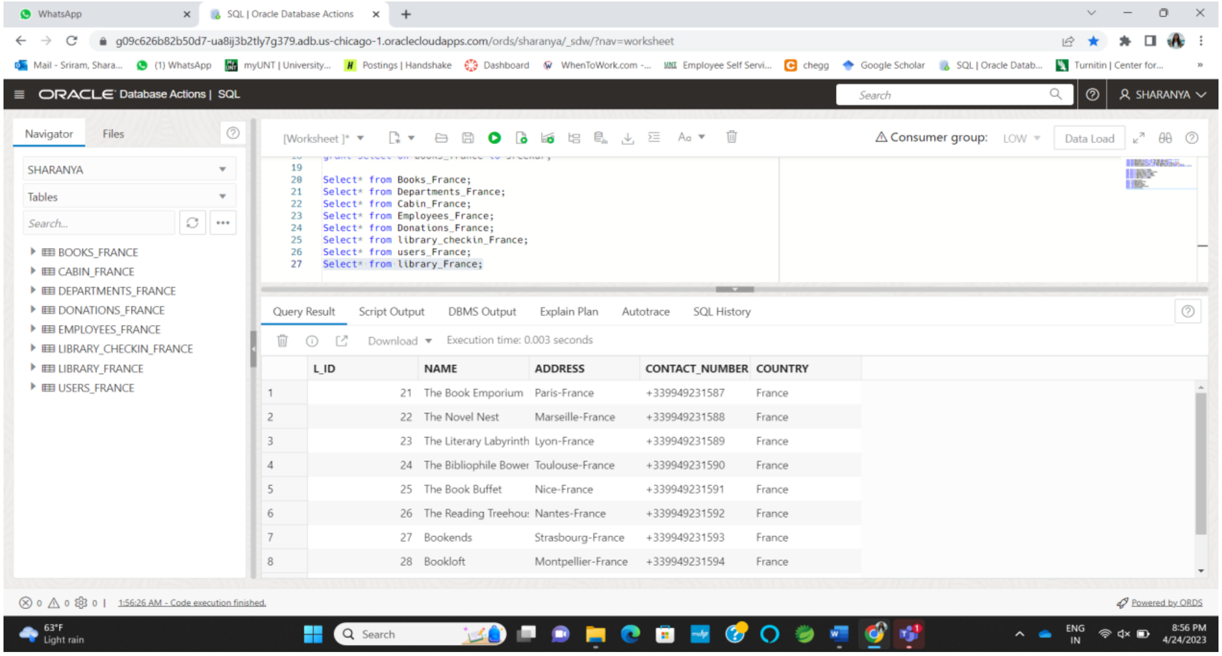
```

Select* from Books_France;
Select* from Departments_France;
Select* from Cabin_France;
Select* from Employees_France;
Select* from Donations_France;
Select* from library_checkin_France;
Select* from users_France;
Select* from library_France;
```

Execution time: 0.005 seconds

	U_ID	U_NAME	ADDRESS	CONTACT	DOB	GENDER	COUNTRY
1	21	Almire	paris	410-715-3997	2/26/1997, 12:00:00 A	F	France
2	22	Harlin	paris	842-652-2557	5/7/1999, 12:00:00 AI	M	France
3	23	Konrad	paris	136-589-7200	8/4/1993, 12:00:00 AI	M	France
4	24	Charmain	paris	446-657-0021	11/19/2005, 12:00:00	F	France
5	25	Tonye	paris	646-729-6890	4/25/1993, 12:00:00 A	F	France
6	26	Roxi	paris	328-912-4275	5/14/1996, 12:00:00 A	F	France
7	27	Christoffer	paris	958-850-2941	4/22/2006, 12:00:00 A	M	France
8	28	Ella	paris	546-671-5748	6/7/1989, 12:00:00 AI	F	France

1:54:35 AM - Code execution finished.

```
Select * from library_France;
```



```
Select* from Books_France;
Select* from Departments_France;
Select* from Cabin_France;
Select* from Employees_France;
Select* from Donations_France;
Select* from library_checkin_France;
Select* from users_France;
Select* from library_France;
```

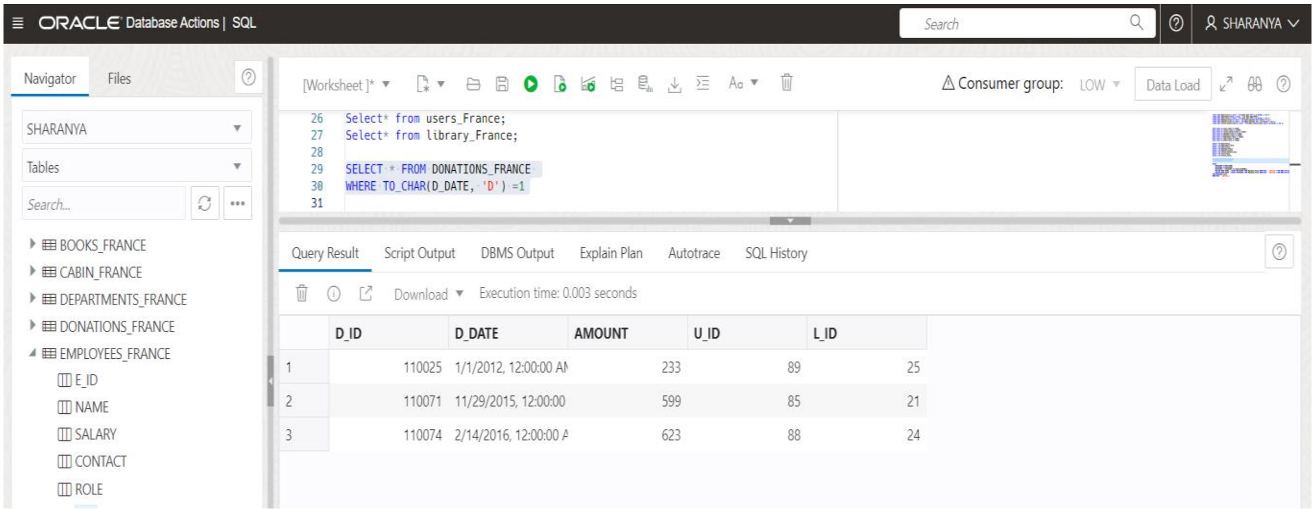
Execution time: 0.003 seconds

	L_ID	NAME	ADDRESS	CONTACT_NUMBER	COUNTRY
1	21	The Book Emporium	Paris-France	+339949231587	France
2	22	The Novel Nest	Marseille-France	+339949231588	France
3	23	The Literary Labyrinth	Lyon-France	+339949231589	France
4	24	The Bibliophile Bower	Toulouse-France	+339949231590	France
5	25	The Book Buffet	Nice-France	+339949231591	France
6	26	The Reading Treehou:	Nantes-France	+339949231592	France
7	27	Bookends	Strasbourg-France	+339949231593	France
8	28	Bookloft	Montpellier-France	+339949231594	France

Working with new scenarios:

Select Queries:

1) Query to Find all the donation details in France that came in January every year.

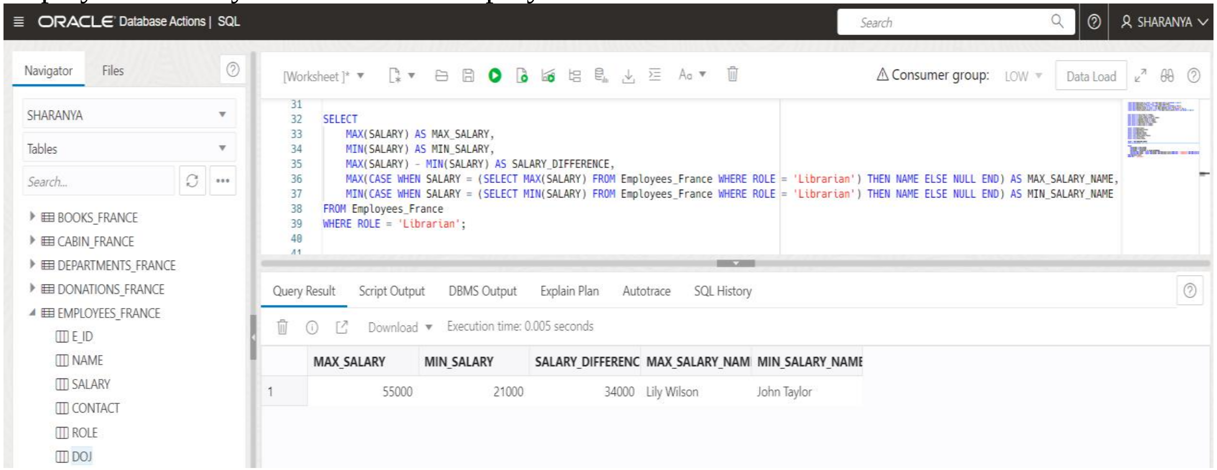


```
Select* from users_France;
Select* from library_France;

SELECT * FROM DONATIONS_FRANCE
WHERE TO_CHAR(D_DATE, 'D') =1
```

Execution time: 0.003 seconds

	D_ID	D_DATE	AMOUNT	U_ID	L_ID
1	110025	1/1/2012, 12:00:00 AI	233	89	25
2	110071	11/29/2015, 12:00:00	599	85	21
3	110074	2/14/2016, 12:00:00 A	623	88	24

2) Query to Find the maximum and minimum salary of Librarian in France employees and display their salary difference and employee names.

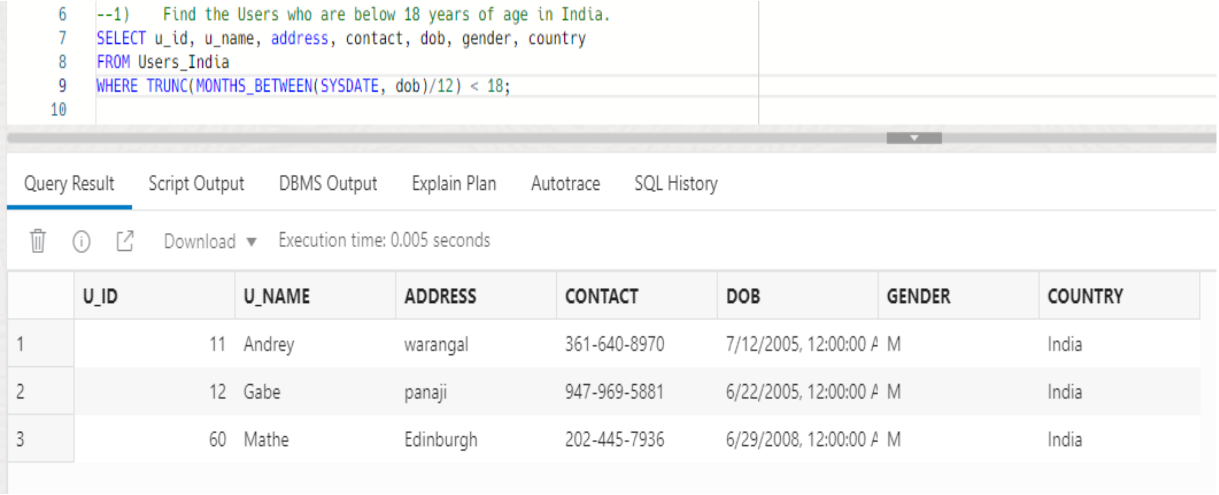


```
SELECT
    MAX(SALARY) AS MAX_SALARY,
    MIN(SALARY) AS MIN_SALARY,
    MAX(SALARY) - MIN(SALARY) AS SALARY_DIFFERENCE,
    MAX(CASE WHEN SALARY = (SELECT MAX(SALARY) FROM Employees_France WHERE ROLE = 'Librarian') THEN NAME ELSE NULL END) AS MAX_SALARY_NAME,
    MIN(CASE WHEN SALARY = (SELECT MIN(SALARY) FROM Employees_France WHERE ROLE = 'Librarian') THEN NAME ELSE NULL END) AS MIN_SALARY_NAME
FROM Employees_France
WHERE ROLE = 'Librarian';
```

Execution time: 0.005 seconds

	MAX_SALARY	MIN_SALARY	SALARY_DIFFERENC	MAX_SALARY_NAM	MIN_SALARY_NAME
1	55000	21000	34000	Lily Wilson	John Taylor

3）Find the Users who are below 18 years of age in India.

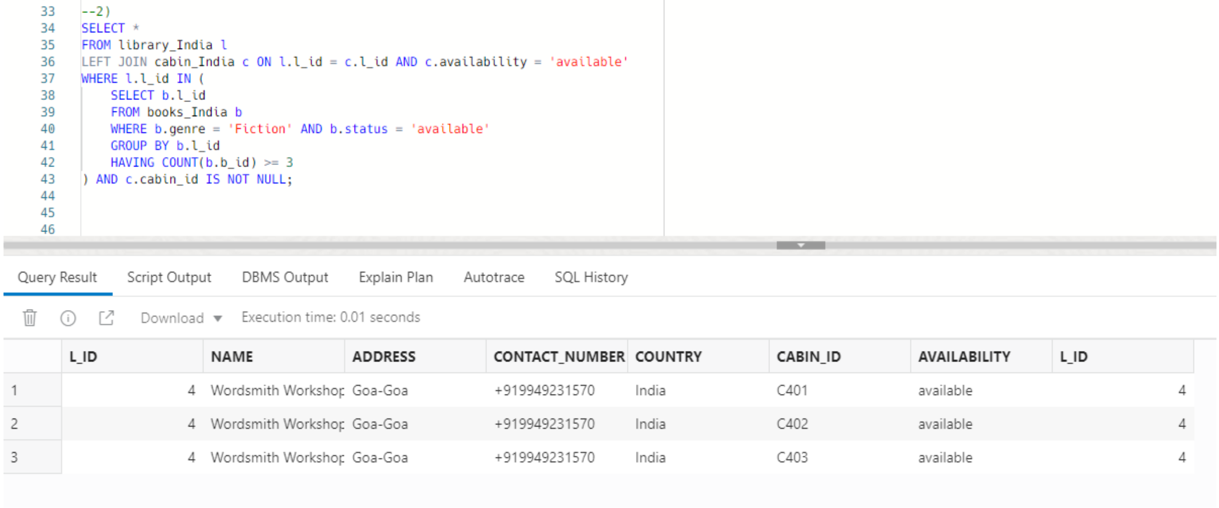


```
--1)    Find the Users who are below 18 years of age in India.
SELECT u_id, u_name, address, contact, dob, gender, country
FROM Users_India
WHERE TRUNC(MONTHS_BETWEEN(SYSDATE, dob)/12) < 18;
```

Execution time: 0.005 seconds

	U_ID	U_NAME	ADDRESS	CONTACT	DOB	GENDER	COUNTRY
1	11	Andrey	warangal	361-640-8970	7/12/2005, 12:00:00 A	M	India
2	12	Gabe	panaji	947-969-5881	6/22/2005, 12:00:00 A	M	India
3	60	Mathe	Edinburgh	202-445-7936	6/29/2008, 12:00:00 A	M	India

4) Retrieve the libraries with that have at least one available cabin and at least 3 available books in the "Fiction" genre.

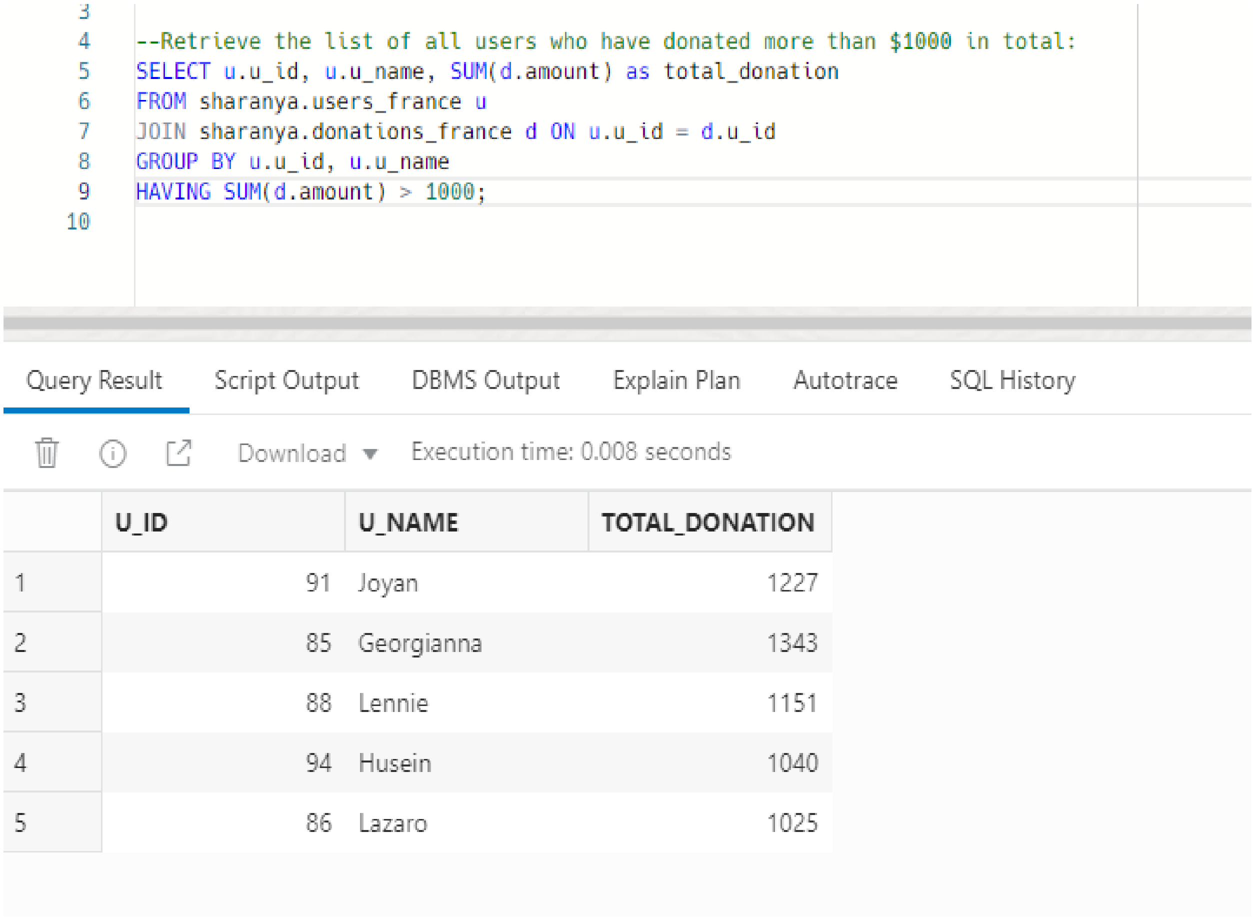


```
--2)
SELECT *
FROM library_India l
LEFT JOIN cabin_India c ON l.l_id = c.l_id AND c.availability = 'available'
WHERE l.l_id IN (
    SELECT b.l_id
    FROM books_India b
    WHERE b.genre = 'Fiction' AND b.status = 'available'
    GROUP BY b.l_id
    HAVING COUNT(b.b_id) >= 3
) AND c.cabin_id IS NOT NULL;
```

Execution time: 0.01 seconds

	L_ID	NAME	ADDRESS	CONTACT_NUMBER	COUNTRY	CABIN_ID	AVAILABILITY	L_ID
1	4	Wordsmith Workshop	Goa-Goa	+919949231570	India	C401	available	4
2	4	Wordsmith Workshop	Goa-Goa	+919949231570	India	C402	available	4
3	4	Wordsmith Workshop	Goa-Goa	+919949231570	India	C403	available	4

5) Retrieve the list of all users who have donated more than $100 in total:

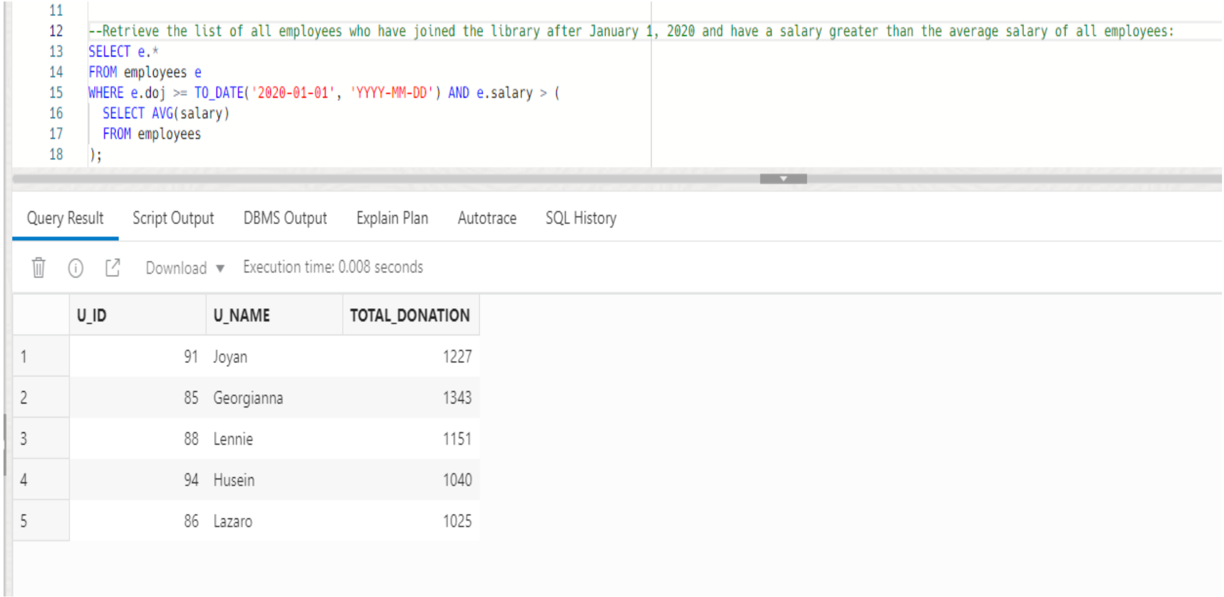


```
--Retrieve the list of all users who have donated more than $1000 in total:
SELECT u.u_id, u.u_name, SUM(d.amount) as total_donation
FROM sharanya.users_france u
JOIN sharanya.donations_france d ON u.u_id = d.u_id
GROUP BY u.u_id, u.u_name
HAVING SUM(d.amount) > 1000;
```

Execution time: 0.008 seconds

	U_ID	U_NAME	TOTAL_DONATION
1	91	Joyan	1227
2	85	Georgianna	1343
3	88	Lennie	1151
4	94	Husein	1040
5	86	Lazaro	1025

6) Retrieve the list of all employees who have joined the library after January 1, 2020 and have a salary greater than the average salary of all employees:

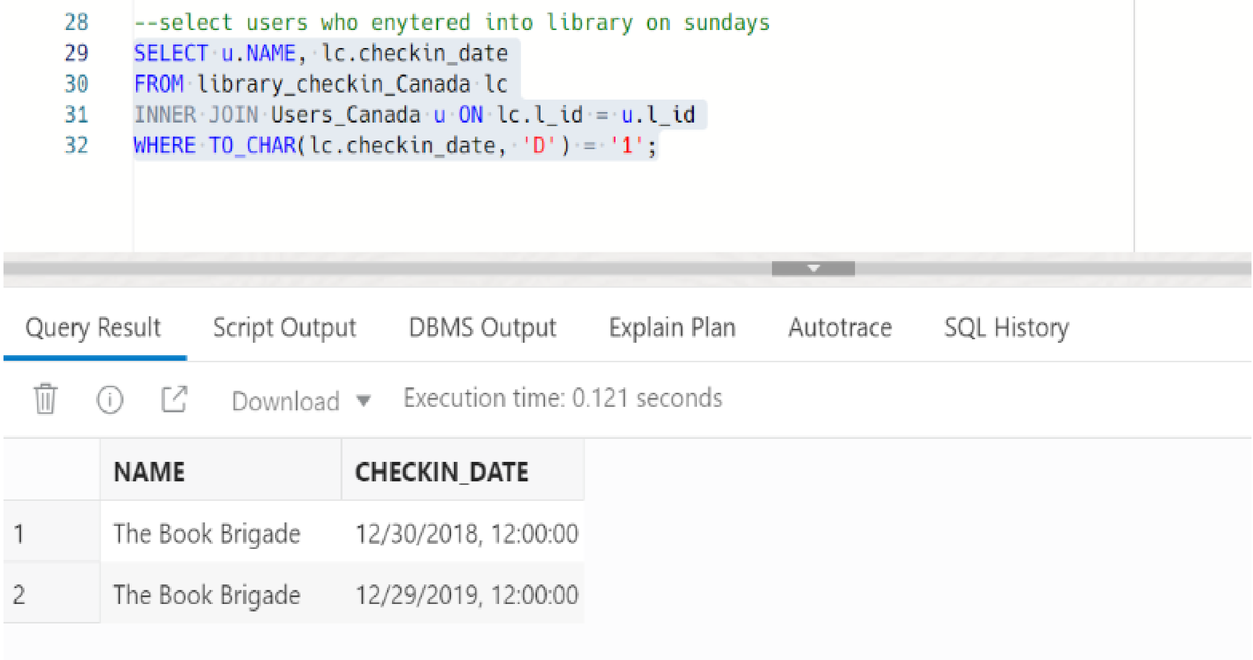


```
--Retrieve the list of all employees who have joined the library after January 1, 2020 and have a salary greater than the average salary of all employees:
SELECT e.*
FROM employees e
WHERE e.doj >= TO_DATE('2020-01-01', 'YYYY-MM-DD') AND e.salary > (
  SELECT AVG(salary)
  FROM employees
);
```

Execution time: 0.008 seconds

	U_ID	U_NAME	TOTAL_DONATION
1	91	Joyan	1227
2	85	Georgianna	1343
3	88	Lennie	1151
4	94	Husein	1040
5	86	Lazaro	1025

7) List all Canada Users who visited library on sundays;

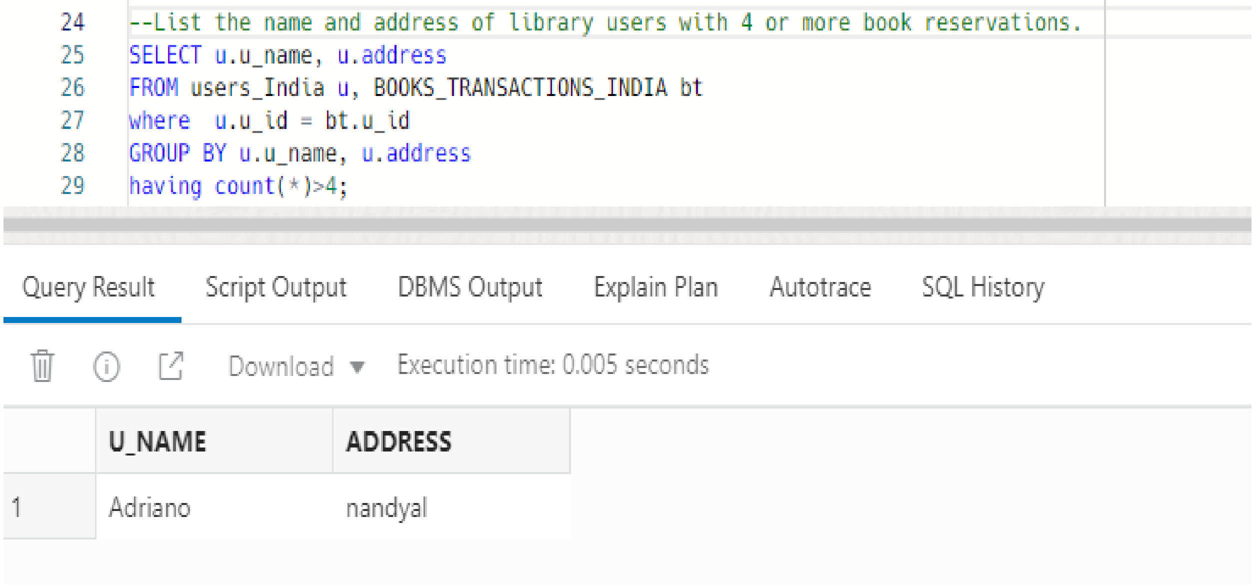


```
--select users who enytered into library on sundays
SELECT u.NAME, lc.checkin_date
FROM library_checkin_Canada lc
INNER JOIN Users_Canada u ON lc.l_id = u.l_id
WHERE TO_CHAR(lc.checkin_date, 'D') = '1';
```

Execution time: 0.121 seconds

	NAME	CHECKIN_DATE
1	The Book Brigade	12/30/2018, 12:00:00
2	The Book Brigade	12/29/2019, 12:00:00

8) List the name and address of library users with 4 or more book reservations;

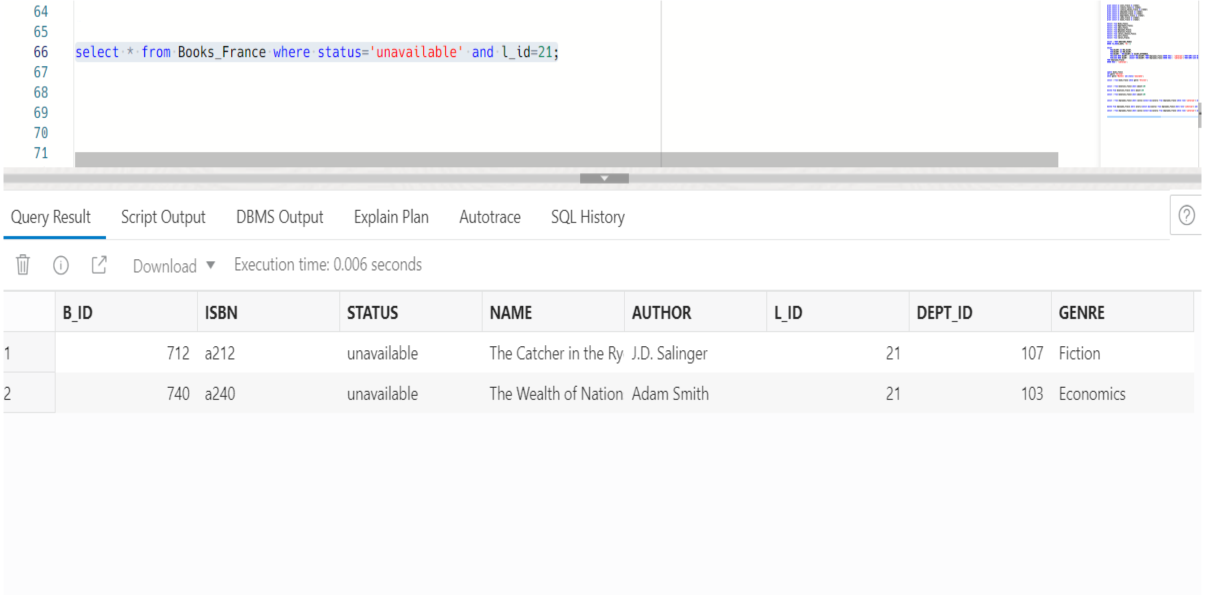


```
--List the name and address of library users with 4 or more book reservations.
SELECT u.u_name, u.address
FROM users_India u, BOOKS_TRANSACTIONS_INDIA bt
where  u.u_id = bt.u_id
GROUP BY u.u_name, u.address
having count(*)>4;
```

Execution time: 0.005 seconds

	U_NAME	ADDRESS
1	Adriano	nandyal

9) List the details of books unavailable in library 21. This query can be applied to all library ids .

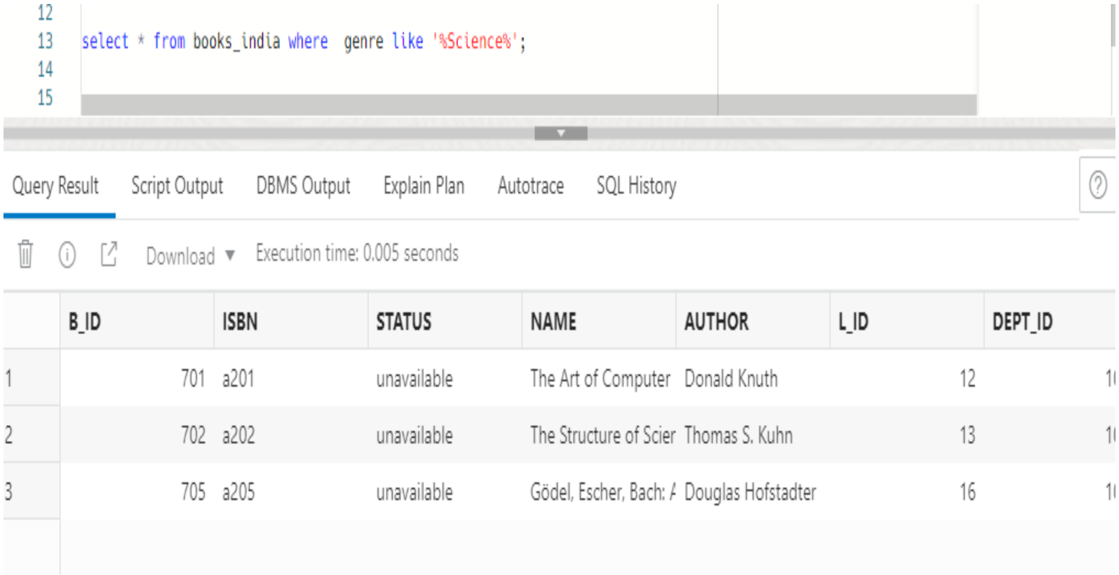


```
select * from Books_France where status='unavailable' and l_id=21;
```

Execution time: 0.006 seconds

	B_ID	ISBN	STATUS	NAME	AUTHOR	L_ID	DEPT_ID	GENRE
1	712	a212	unavailable	The Catcher in the Ry	J.D. Salinger	21	107	Fiction
2	740	a240	unavailable	The Wealth of Nation	Adam Smith	21	103	Economics

10) Finding all the Science Genre books from India



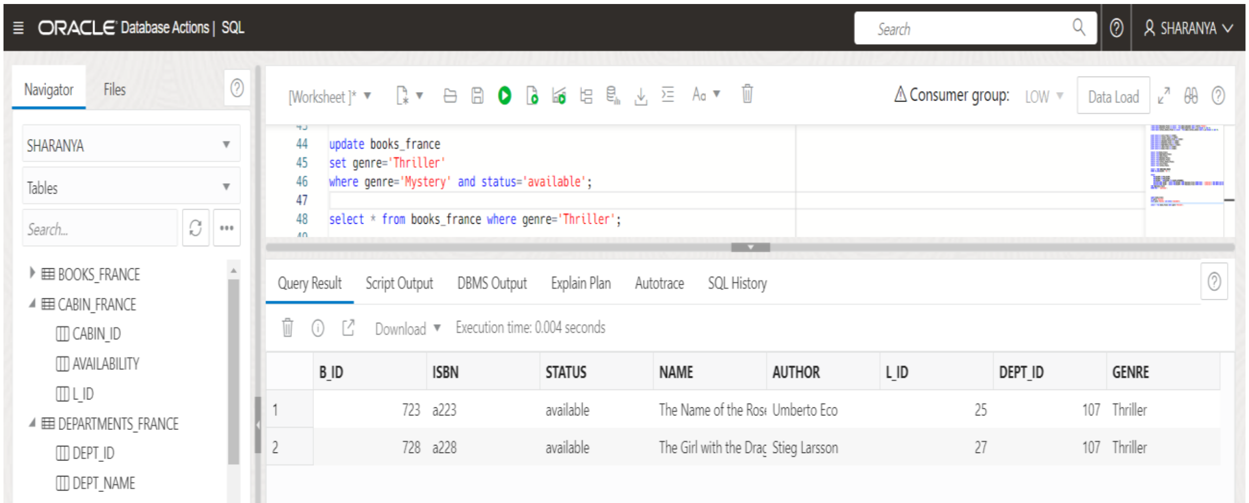
```
select * from books_india where  genre like '%Science%';
```

Execution time: 0.005 seconds

	B_ID	ISBN	STATUS	NAME	AUTHOR	L_ID	DEPT_ID
1	701	a201	unavailable	The Art of Computer	Donald Knuth	12	1
2	702	a202	unavailable	The Structure of Scier	Thomas S. Kuhn	13	1
3	705	a205	unavailable	Gödel, Escher, Bach: A	Douglas Hofstadter	16	1

Update queries:

1)Here we are updating the all the mystery genre books to Thriller if they are available in the library. This is done in the France site of the library.

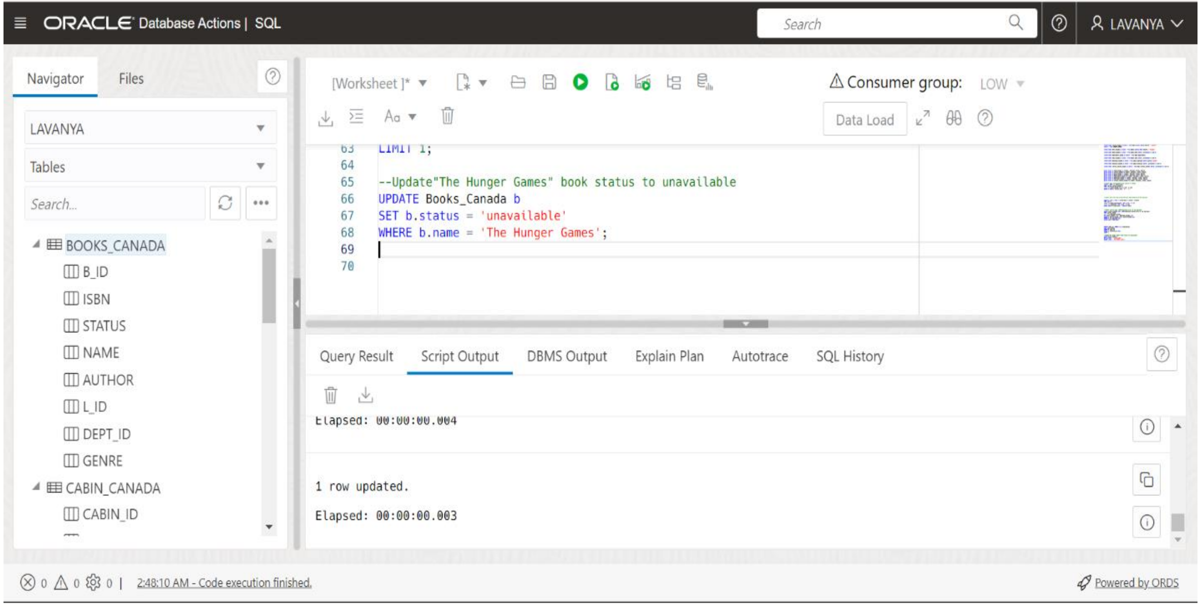


```
update books_france
set genre='Thriller'
where genre='Mystery' and status='available';

select * from books_france where genre='Thriller';
```

Execution time: 0.004 seconds

	B_ID	ISBN	STATUS	NAME	AUTHOR	L_ID	DEPT_ID	GENRE
1	723	a223	available	The Name of the Ros	Umberto Eco	25	107	Thriller
2	728	a228	available	The Girl with the Drag	Stieg Larsson	27	107	Thriller

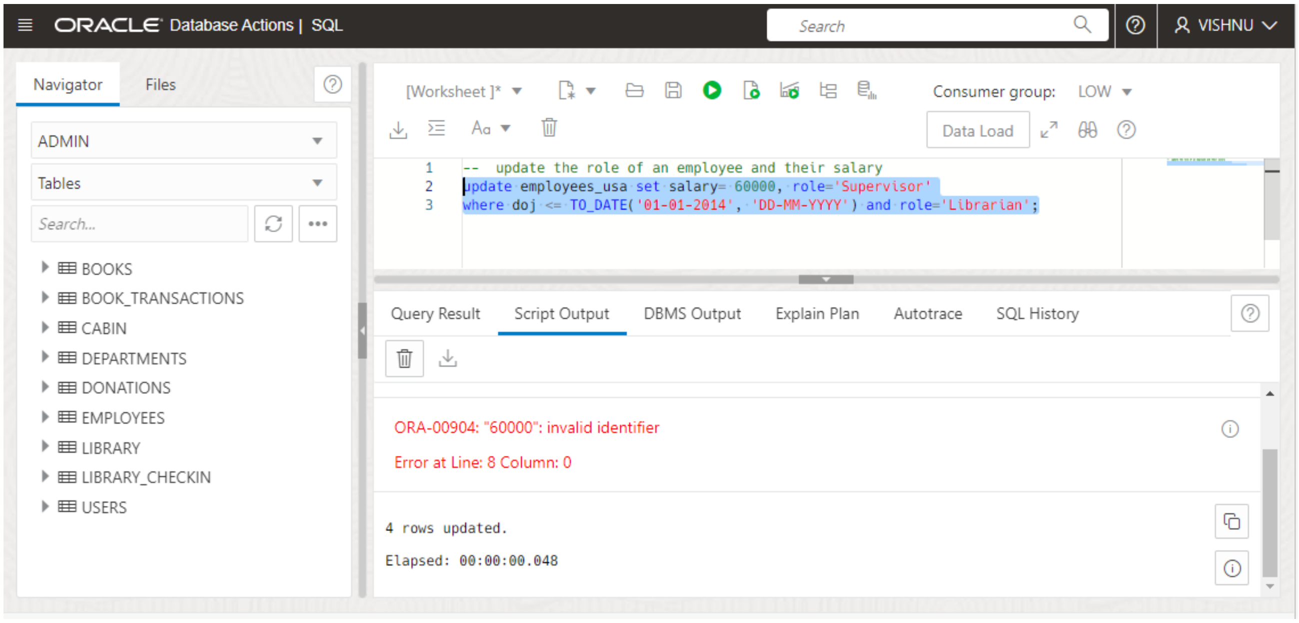
2. Update “The Hunger Games” book status to unavailable.



```
LIMIT 1;

--Update"The Hunger Games" book status to unavailable
UPDATE Books_Canada b
SET b.status = 'unavailable'
WHERE b.name = 'The Hunger Games';

```

```
Elapsed: 00:00:00.004


1 row updated.

Elapsed: 00:00:00.003
```

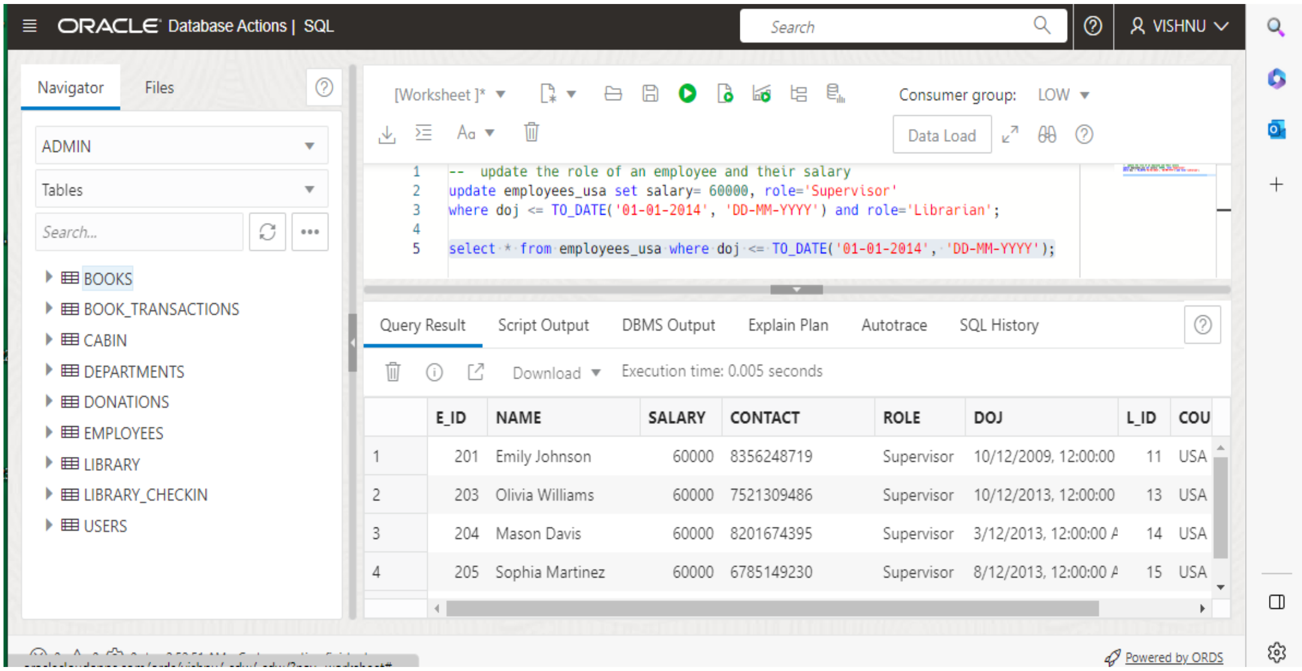
3) Gave promotion to librarians who are working from before 2014 and updated their salary to 60000



```sql
--  update the role of an employee and their salary
update employees_usa set salary= 60000, role='Supervisor'
where doj <= TO_DATE('01-01-2014', 'DD-MM-YYYY') and role='Librarian';
```

```
ORA-00904: "60000": invalid identifier

Error at Line: 8 Column: 0

4 rows updated.

Elapsed: 00:00:00.048
```

Data is as follows

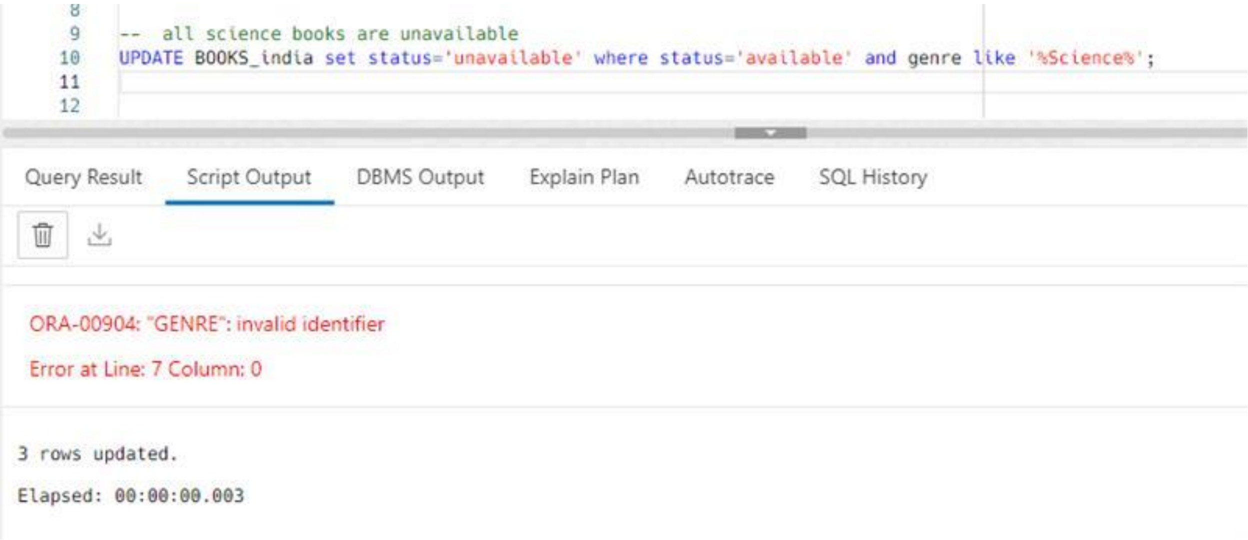


```sql
--  update the role of an employee and their salary
update employees_usa set salary= 60000, role='Supervisor'
where doj <= TO_DATE('01-01-2014', 'DD-MM-YYYY') and role='Librarian';

select * from employees_usa where doj <= TO_DATE('01-01-2014', 'DD-MM-YYYY');
```

Execution time: 0.005 seconds

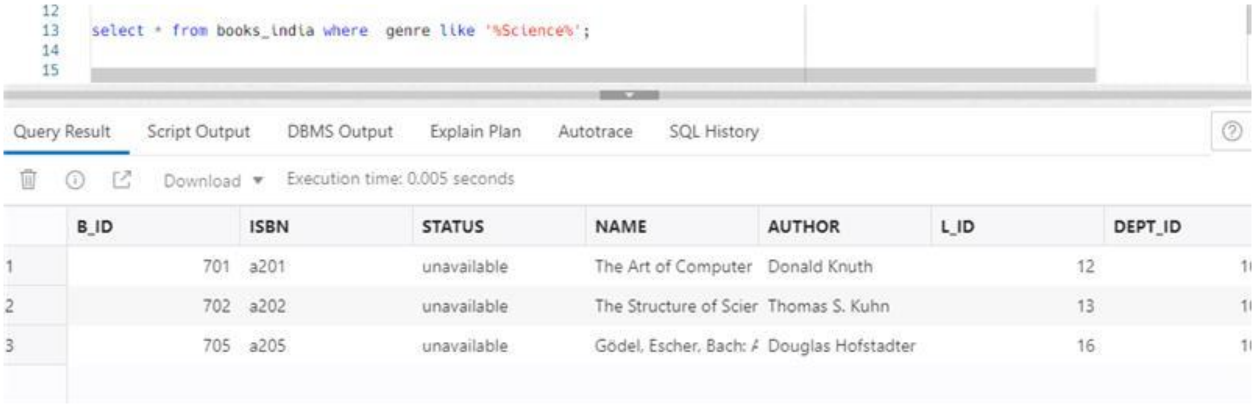
	E_ID	NAME	SALARY	CONTACT	ROLE	DOJ	L_ID	COU
1	201	Emily Johnson	60000	8356248719	Supervisor	10/12/2009, 12:00:00	11	USA
2	203	Olivia Williams	60000	7521309486	Supervisor	10/12/2013, 12:00:00	13	USA
3	204	Mason Davis	60000	8201674395	Supervisor	3/12/2013, 12:00:00 A	14	USA
4	205	Sophia Martinez	60000	6785149230	Supervisor	8/12/2013, 12:00:00 A	15	USA

4）All the science books are reserved by the user and the the books are made unavailabe for other users



```
-- all science books are unavailable
UPDATE BOOKS_india set status='unavailable' where status='available' and genre like '%Science%';
```

```
ORA-00904: "GENRE": invalid identifier
Error at Line: 7 Column: 0

3 rows updated.
Elapsed: 00:00:00.003
```

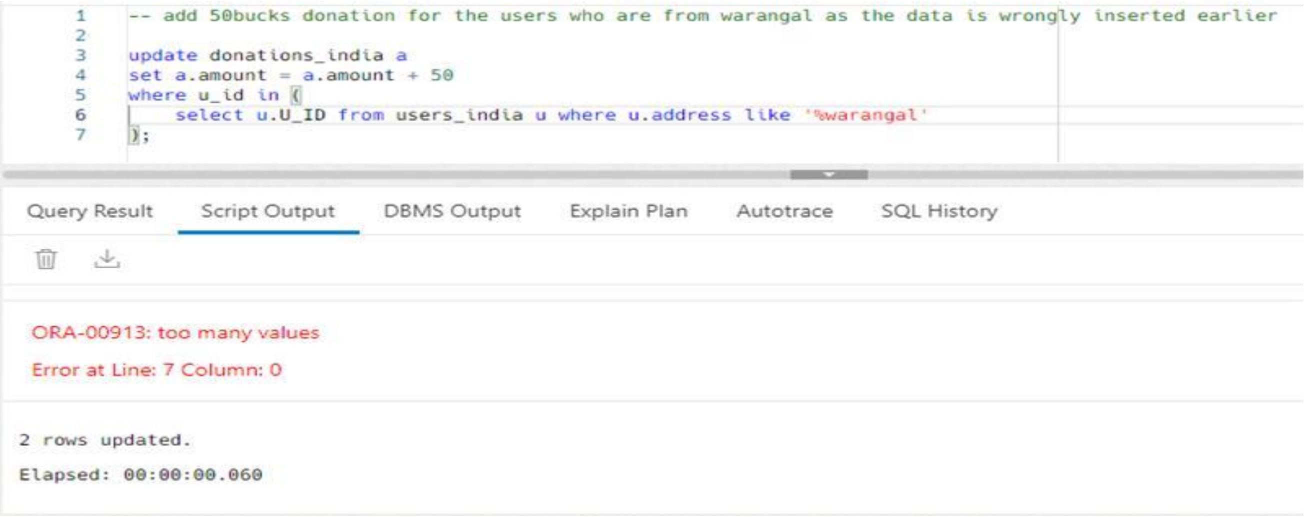


```
select * from books_india where  genre like '%Science%';
```

Execution time: 0.005 seconds

	B_ID	ISBN	STATUS	NAME	AUTHOR	L_ID	DEPT_ID
1	701	a201	unavailable	The Art of Computer	Donald Knuth	12	1
2	702	a202	unavailable	The Structure of Scier	Thomas S. Kuhn	13	1
3	705	a205	unavailable	Gödel, Escher, Bach: A	Douglas Hofstadter	16	1

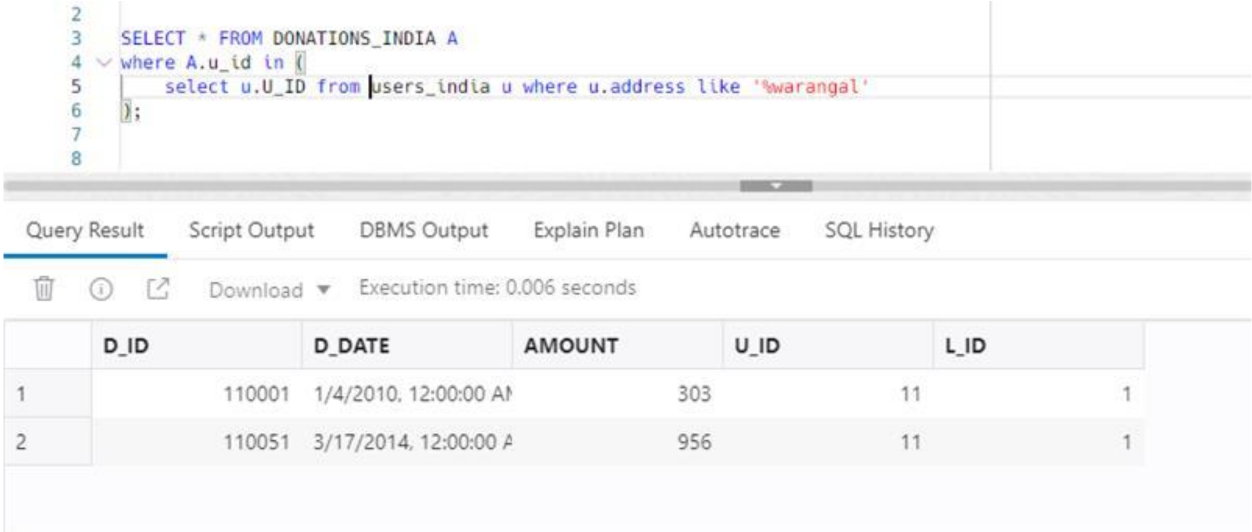
5) Adding 50bucks donation for the users who are from warangal as the data is wrongly inserted.



```
-- add 50bucks donation for the users who are from warangal as the data is wrongly inserted earlier

update donations_india a
set a.amount = a.amount + 50
where u_id in (
    select u.U_ID from users_india u where u.address like '%warangal'
);
```

```
ORA-00913: too many values
Error at Line: 7 Column: 0

2 rows updated.
Elapsed: 00:00:00.060
```

RESULT JUSTIFYING ABOVE QUERY. SHOWING THAT THE USER FROM WARANGAL HAS DONATED AMOUNT

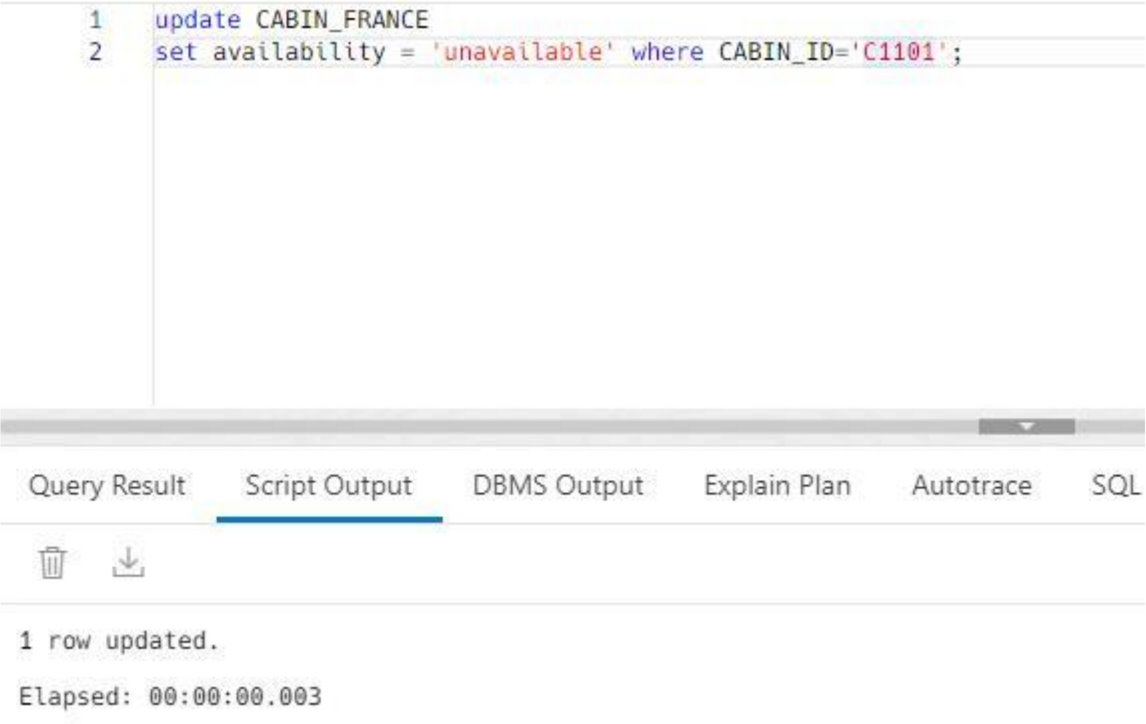


```
SELECT * FROM DONATIONS_INDIA A
where A.u_id in (
    select u.U_ID from users_india u where u.address like '%warangal'
);
```

Execution time: 0.006 seconds

	D_ID	D_DATE	AMOUNT	U_ID	L_ID
1	110001	1/4/2010, 12:00:00 AI	303	11	1
2	110051	3/17/2014, 12:00:00 A	956	11	1

6) USER RESERVED A CABIN

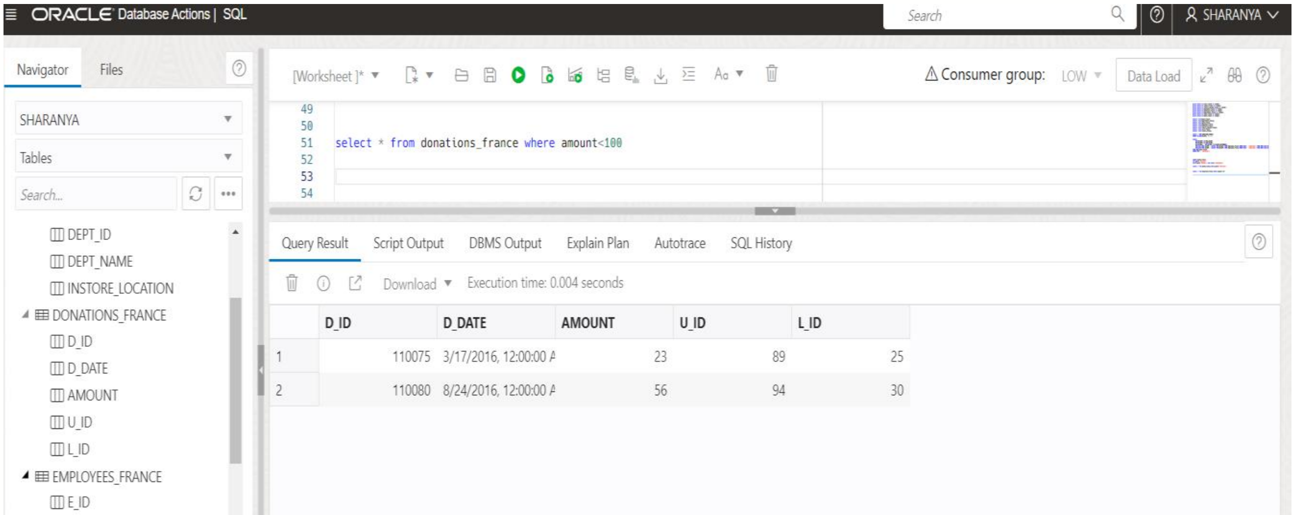


```
update CABIN_FRANCE
set availability = 'unavailable' where CABIN_ID='C1101';
```

```
1 row updated.

Elapsed: 00:00:00.003
```

Delete queries:

1) Updating the database so that it stores the donations data if and only if the amount exceeds 100 dollars.

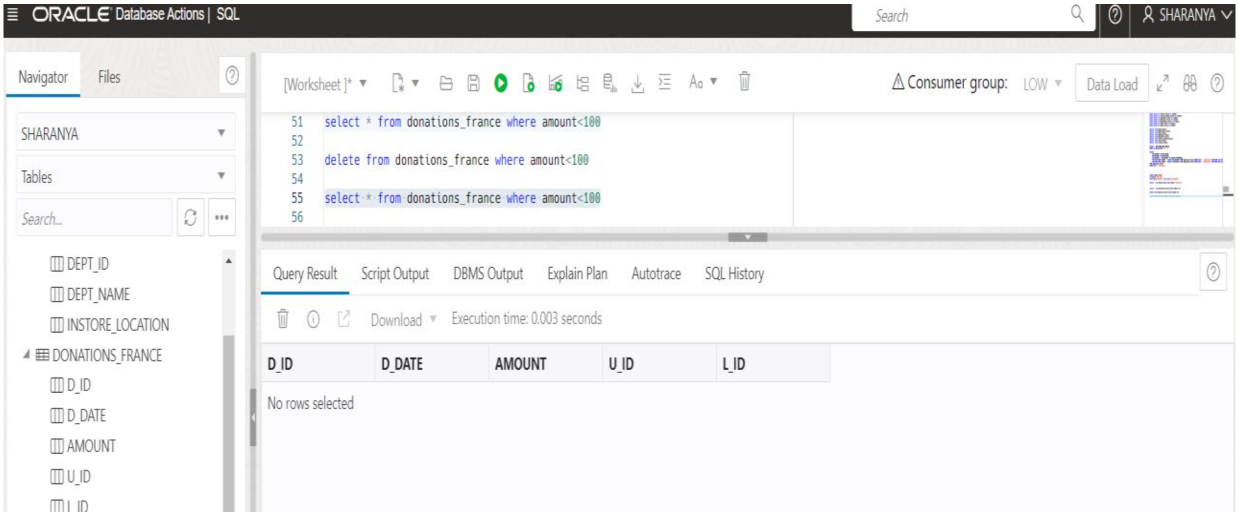


```
select * from donations_france where amount<100
```

Execution time: 0.004 seconds

	D_ID	D_DATE	AMOUNT	U_ID	L_ID
1	110075	3/17/2016, 12:00:00 A	23	89	25
2	110080	8/24/2016, 12:00:00 A	56	94	30

Data before the donations are deleted.



```
select * from donations_france where amount<100

delete from donations_france where amount<100

select * from donations_france where amount<100
```

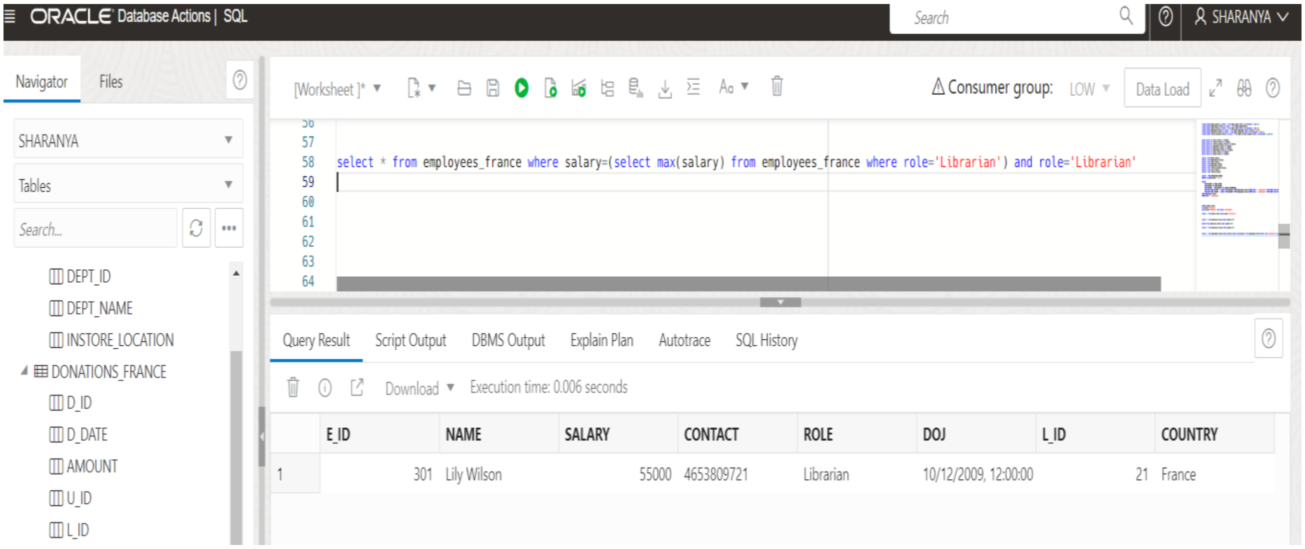
Execution time: 0.003 seconds

D_ID	D_DATE	AMOUNT	U_ID	L_ID

No rows selected

Data after the donations are deleted.

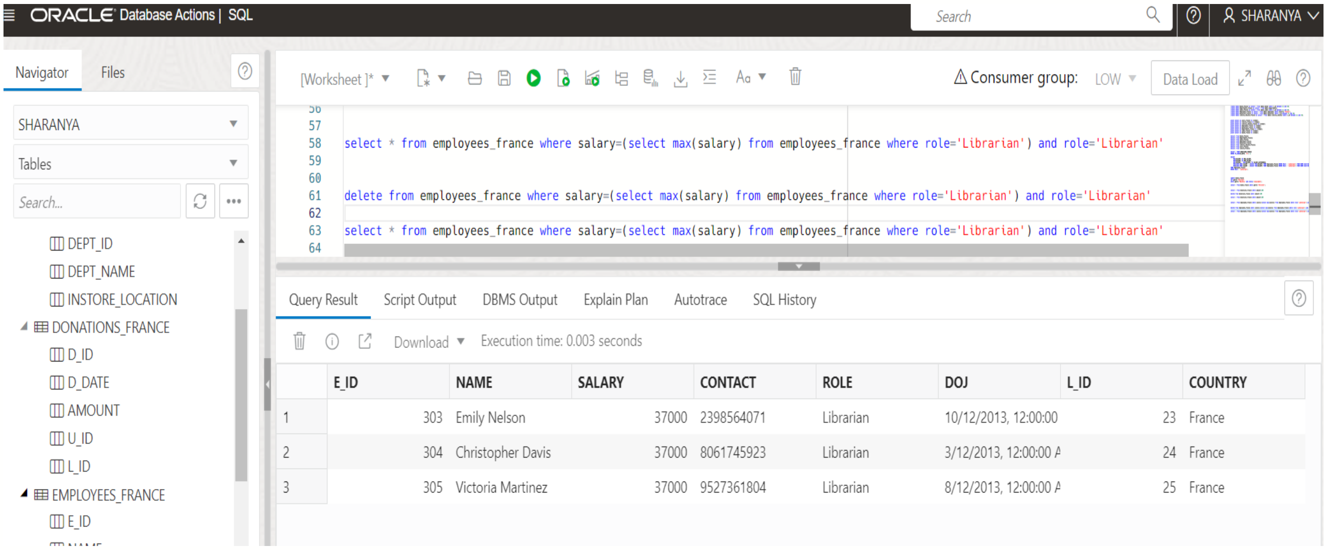
2)Data before deletion. Here we can see that the librarian with highest salary is Lily and her salary is 55000.



```
select * from employees_france where salary=(select max(salary) from employees_france where role='Librarian') and role='Librarian'
```

	E_ID	NAME	SALARY	CONTACT	ROLE	DOJ	L_ID	COUNTRY
1	301	Lily Wilson	55000	4653809721	Librarian	10/12/2009, 12:00:00	21	France

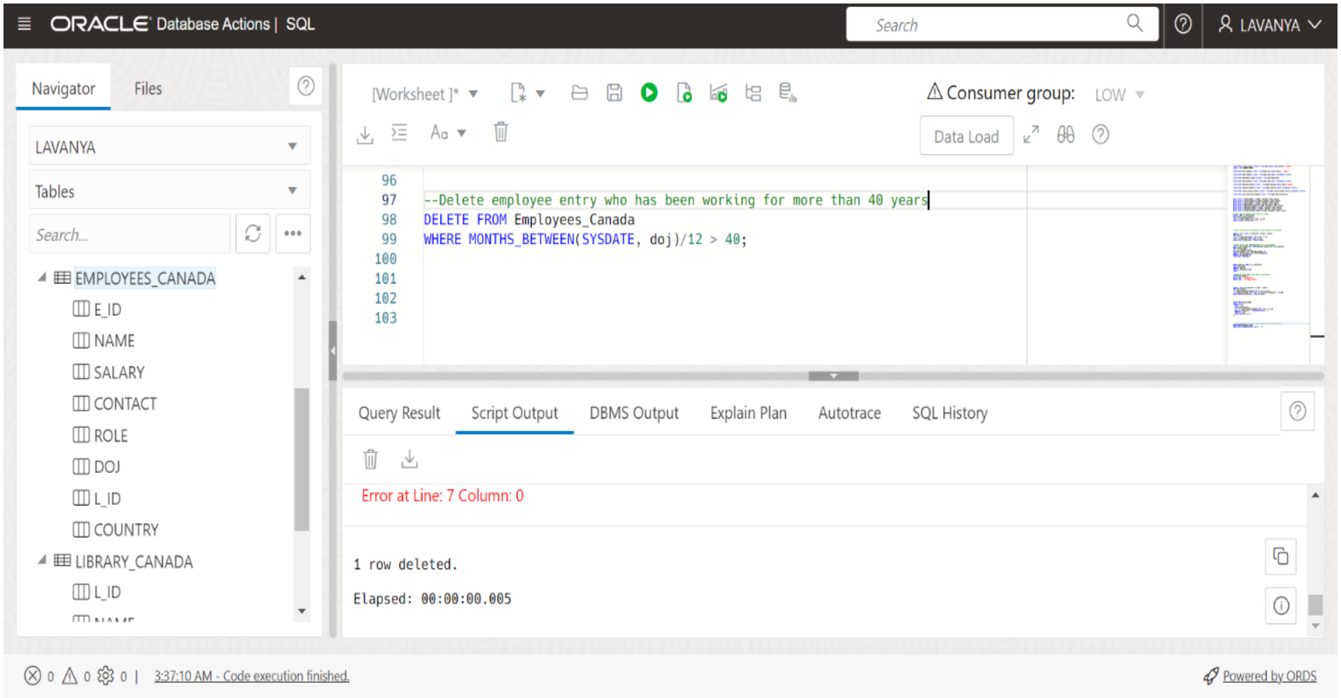
Data after deletion. Here we can see that librarian has highest salary of 37000 and data of Lily is not present as she is fired.



```
select * from employees_france where salary=(select max(salary) from employees_france where role='Librarian') and role='Librarian'


delete from employees_france where salary=(select max(salary) from employees_france where role='Librarian') and role='Librarian'

select * from employees_france where salary=(select max(salary) from employees_france where role='Librarian') and role='Librarian'
```

	E_ID	NAME	SALARY	CONTACT	ROLE	DOJ	L_ID	COUNTRY
1	303	Emily Nelson	37000	2398564071	Librarian	10/12/2013, 12:00:00	23	France
2	304	Christopher Davis	37000	8061745923	Librarian	3/12/2013, 12:00:00 A	24	France
3	305	Victoria Martinez	37000	9527361804	Librarian	8/12/2013, 12:00:00 A	25	France

3) Delete employee who has been working for more than 40 years.

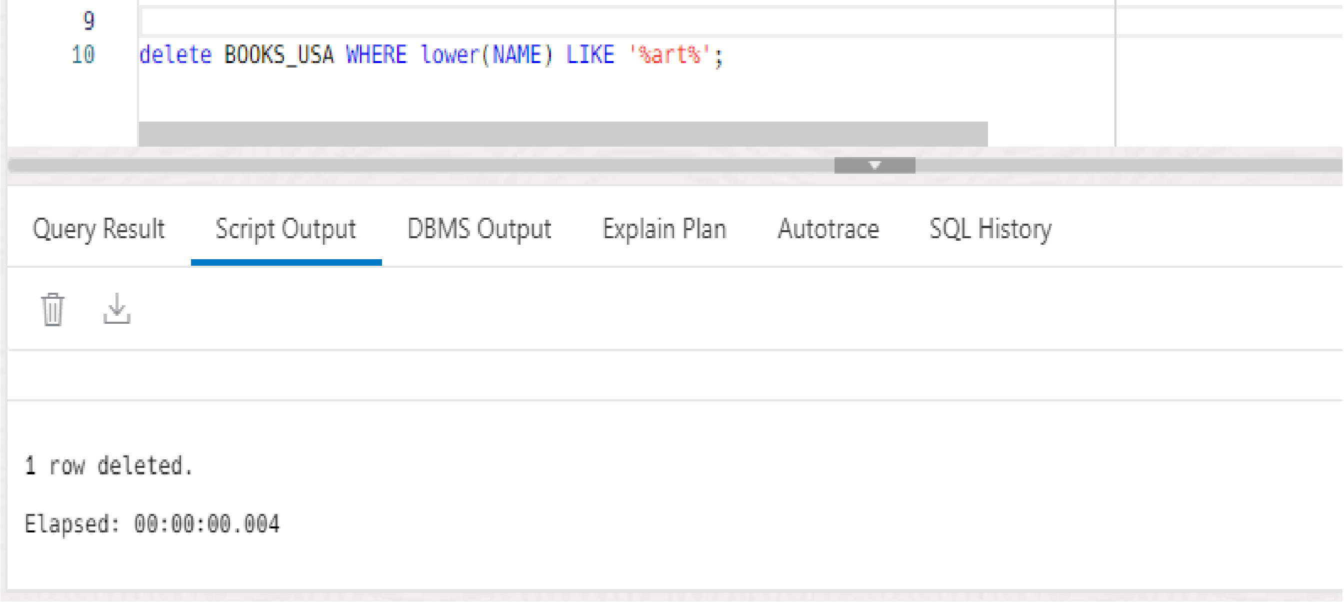


```
--Delete employee entry who has been working for more than 40 years
DELETE FROM Employees_Canada
WHERE MONTHS_BETWEEN(SYSDATE, doj)/12 > 40;
```

Error at Line: 7 Column: 0

```
1 row deleted.

Elapsed: 00:00:00.005
```

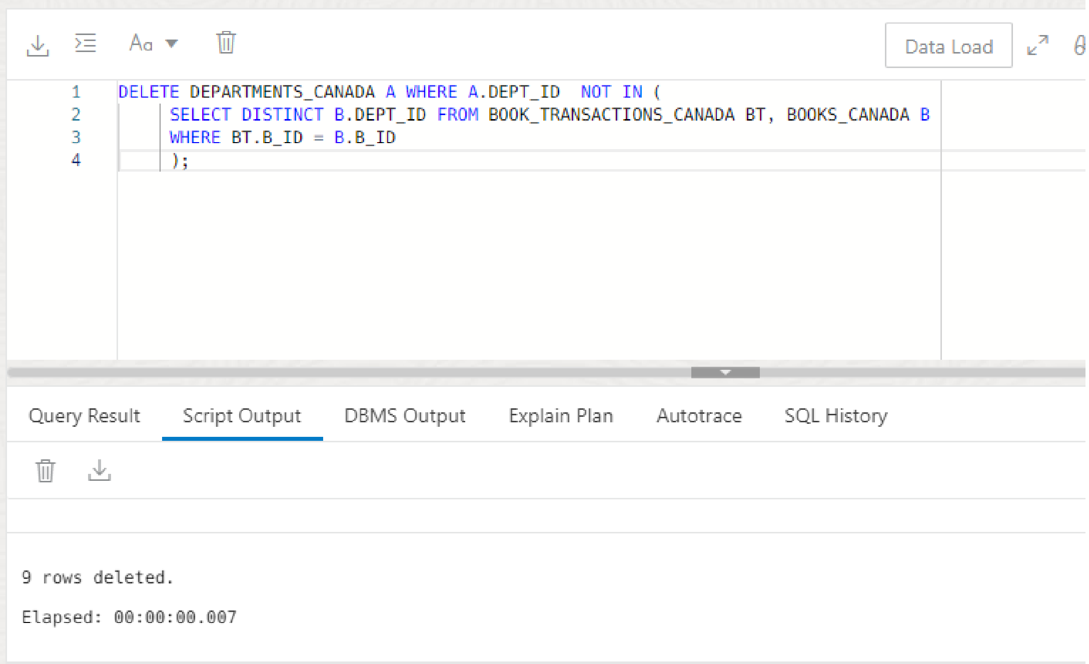
4)Deleted books having title as ART from USA location.



```
delete BOOKS_USA WHERE lower(NAME) LIKE '%art%';
```

```
1 row deleted.

Elapsed: 00:00:00.004
```

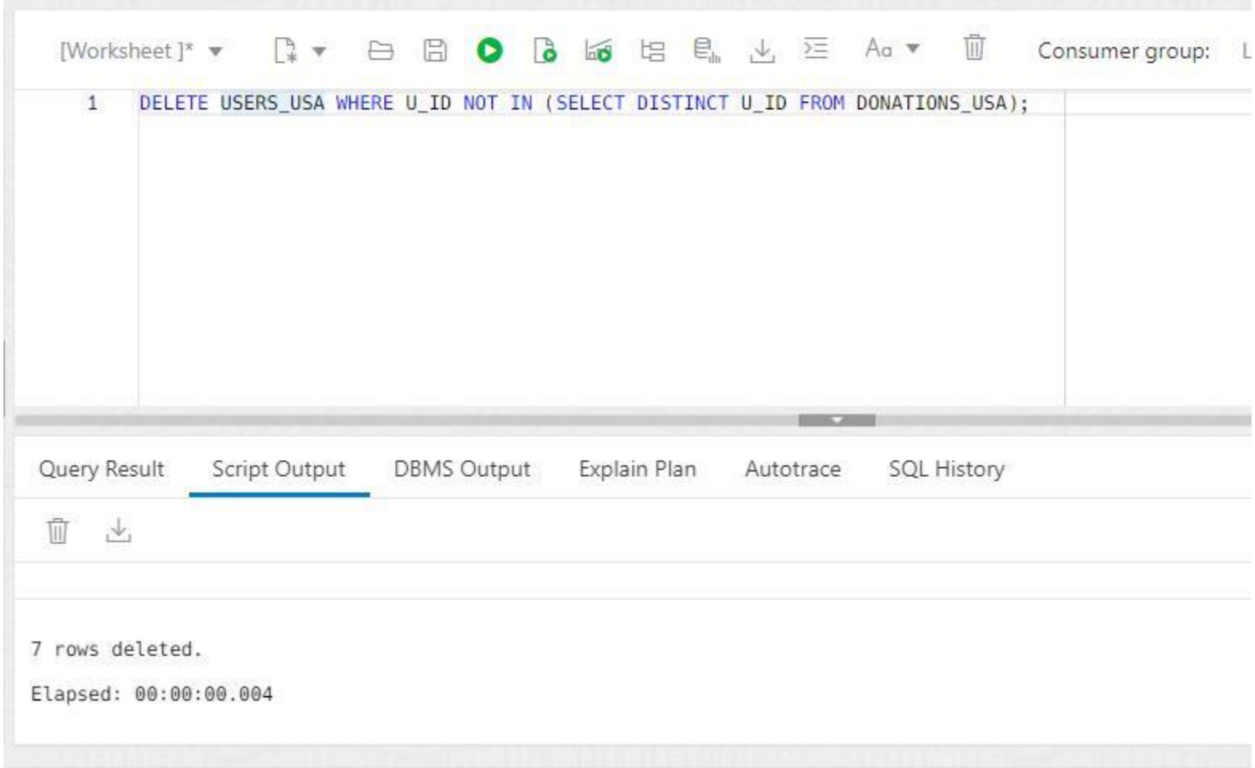
5) DELETED THE DEPARTMENTS THAT ARE NOT IN USE



```
DELETE DEPARTMENTS_CANADA A WHERE A.DEPT_ID  NOT IN (
    SELECT DISTINCT B.DEPT_ID FROM BOOK_TRANSACTIONS_CANADA BT, BOOKS_CANADA B
    WHERE BT.B_ID = B.B_ID
    );
```

```
9 rows deleted.

Elapsed: 00:00:00.007
```

6) DELETE USERS WHO DID NOT DO ANY DONATIONS

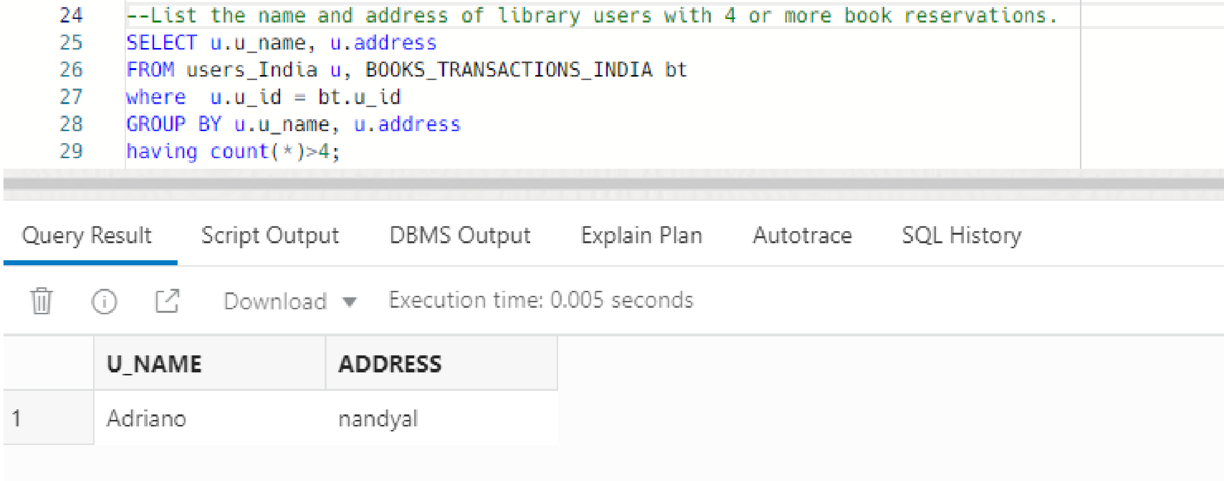


```
DELETE USERS_USA WHERE U_ID NOT IN (SELECT DISTINCT U_ID FROM DONATIONS_USA);
```

```
7 rows deleted.

Elapsed: 00:00:00.004
```

Queries By the Professor :

1) List the name and address of library users with 4 or more book reservations;

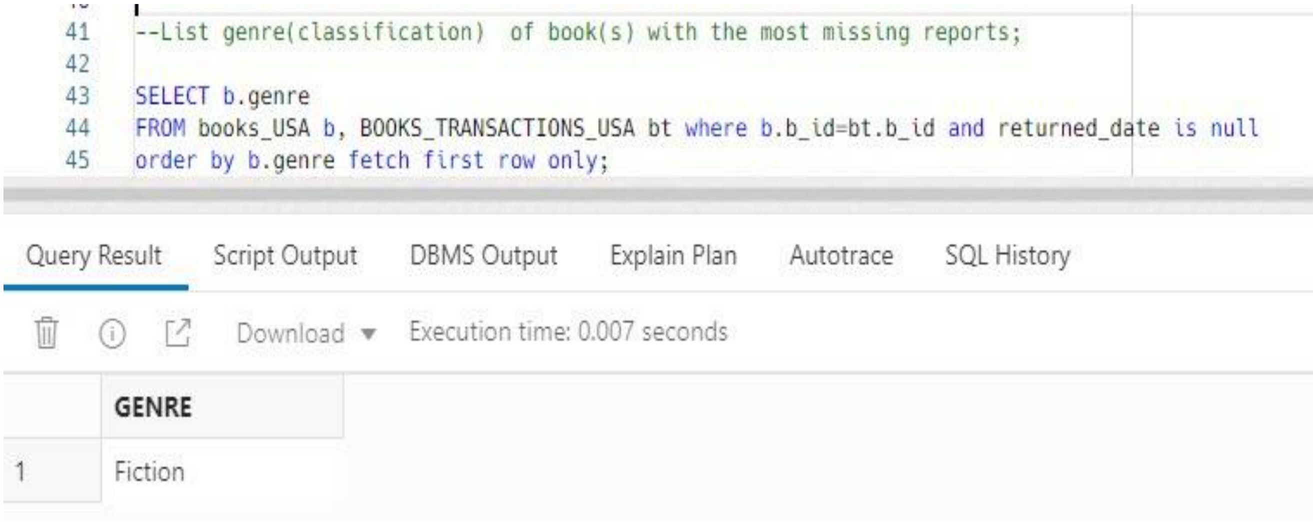


```
--List the name and address of library users with 4 or more book reservations.
SELECT u.u_name, u.address
FROM users_India u, BOOKS_TRANSACTIONS_INDIA bt
where  u.u_id = bt.u_id
GROUP BY u.u_name, u.address
having count(*)>4;
```

Execution time: 0.005 seconds

	U_NAME	ADDRESS
1	Adriano	nandyal

2) List genre(classification of books with the most missing reports.

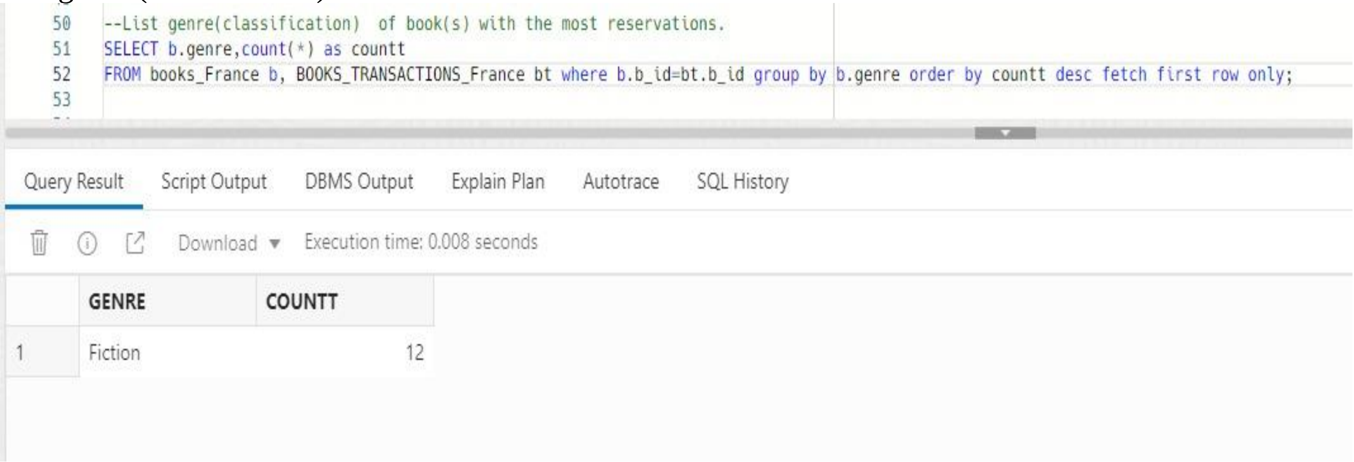


```
--List genre(classification)  of book(s) with the most missing reports;

SELECT b.genre
FROM books_USA b, BOOKS_TRANSACTIONS_USA bt where b.b_id=bt.b_id and returned_date is null
order by b.genre fetch first row only;
```

Execution time: 0.007 seconds

	GENRE
1	Fiction

3) List genre (classification) of books with the most reservations.

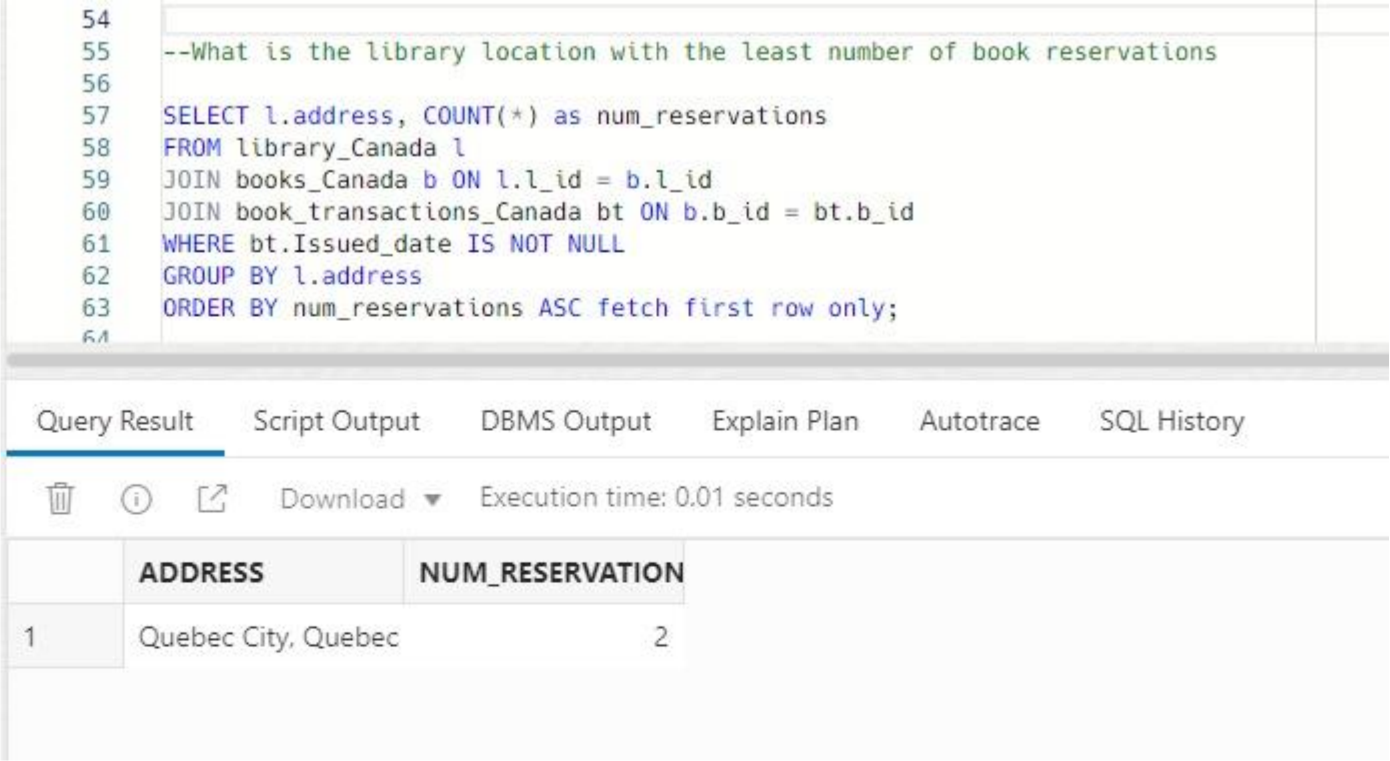


```
--List genre(classification)  of book(s) with the most reservations.
SELECT b.genre,count(*) as count
FROM books_France b, BOOKS_TRANSACTIONS_France bt where b.b_id=bt.b_id group by b.genre order by countt desc fetch first row only;
```

Execution time: 0.008 seconds

	GENRE	COUNTT
1	Fiction	12

4) What is the library location with the lease number of book reservations.

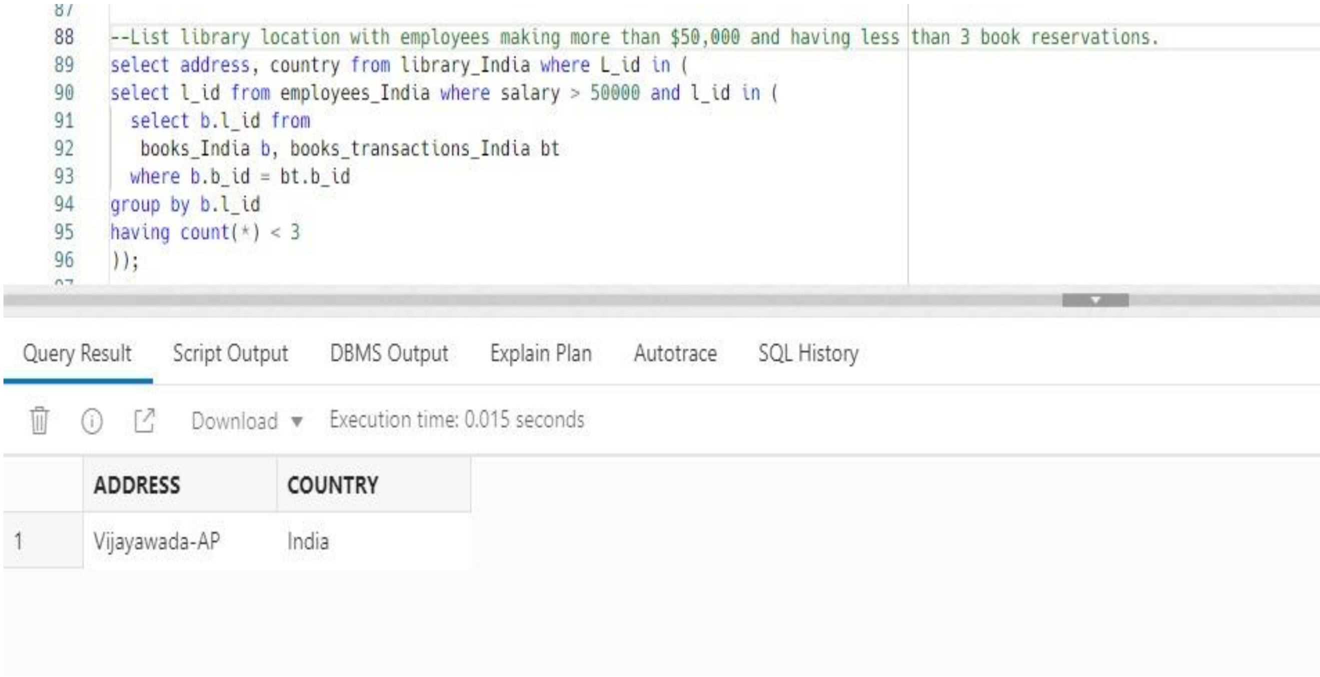


```
--What is the library location with the least number of book reservations

SELECT l.address, COUNT(*) as num_reservations
FROM library_Canada l
JOIN books_Canada b ON l.l_id = b.l_id
JOIN book_transactions_Canada bt ON b.b_id = bt.b_id
WHERE bt.Issued_date IS NOT NULL
GROUP BY l.address
ORDER BY num_reservations ASC fetch first row only;
```

Execution time: 0.01 seconds

	ADDRESS	NUM_RESERVATION
1	Quebec City, Quebec	2

5) List library location with employees making more than $50000 and having less than 3 book reservations.

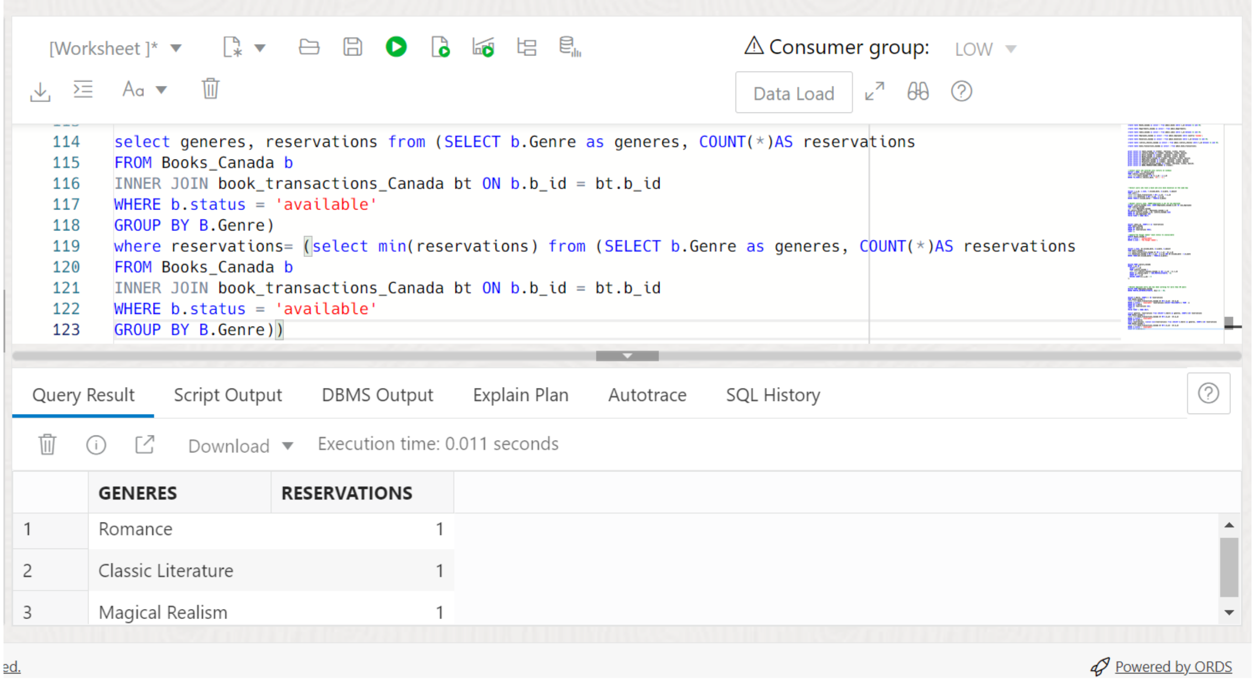


```
--List library location with employees making more than $50,000 and having less than 3 book reservations.
select address, country from library_India where L_id in (
select l_id from employees_India where salary > 50000 and l_id in (
  select b.l_id from
   books_India b, books_transactions_India bt
  where b.b_id = bt.b_id
group by b.l_id
having count(*) < 3
));
```

Execution time: 0.015 seconds

	ADDRESS	COUNTRY
1	Vijayawada-AP	India

6) List the count of number of reservations for each genre in the Books Canada Region.



```
select generes, reservations from (SELECT b.Genre as generes, COUNT(*)AS reservations
FROM Books_Canada b
INNER JOIN book_transactions_Canada bt ON b.b_id = bt.b_id
WHERE b.status = 'available'
GROUP BY B.Genre)
where reservations= (select min(reservations) from (SELECT b.Genre as generes, COUNT(*)AS reservations
FROM Books_Canada b
INNER JOIN book_transactions_Canada bt ON b.b_id = bt.b_id
WHERE b.status = 'available'
GROUP BY B.Genre))
```

Execution time: 0.011 seconds

	GENERES	RESERVATIONS
1	Romance	1
2	Classic Literature	1
3	Magical Realism	1

Individual contribution:

I am the admin of the cloud Database, I have created users to all my teammates and to myself as well.

In part 4 of project, I am the owner of site 1, which holds all the data for India. I have fragmented the data of India on to my site from the global database. I have performed 2 select queries and 1 update query and 1 delete query and 3 queries from the list of queries provided by professor.