

PROJECT REPORT

CAMP BI V3

BATCH 1 : GROUP 1

INDEX

<u>S. No.</u>	<u>Contents</u>	<u>Page No.</u>
O1.	Project Description	02
02.	Unit Test Report	02 - 09
	a Creation of Database	02
	b Creation of Schemas	03
	c Creation of Tables as per Dataset	03
	d Creation of Integration Object	04
	e Creation of External Stage for loading the data structure	04
	f Creation of Stream on the given table	05
	g Creation of Snowpipe for auto-ingestion of data from S3 bucket	05
	h SCD operations on the Consumer Table	06
03.	Data Analysis on given Dataset	07-09

Supervised By

Mr. Pritam Gorain

Group Members :

- Mr. Aadarsh Awasthi
- Mr. Akshat Dwivedi
- Mr. Kumar Naman
- Mr. Siddharth Rai
- Ms. Tulsi Gupta
- Mr. Utkarsh Rathore
- Mr. Vishal Gaur

PROJECT DESCRIPTION

The project involves data ingestion and analysis from public datahub Kaggle [Link](#).

- Steps involved in performing the data ingestion:
 - a. Loading data to external stage
 - b. Ingesting data into the landing schema
 - c. Ingesting data into the consumer table
 - d. Perform analysis on the given dataset

UNIT TEST REPORT

1. Created a database named **SF_PROJECT**;

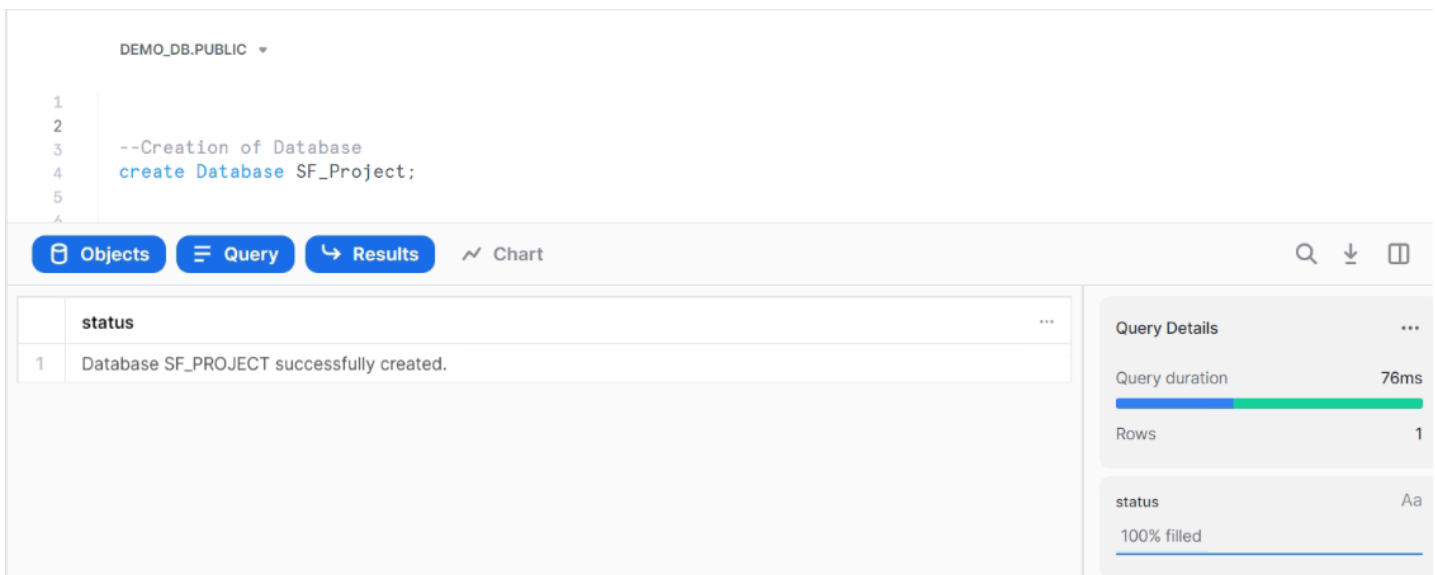


Fig: 01

2. Created three schemas named **ITR_RDS**, **ITR_RDS_LANDING** and **ITR_DIS**

The screenshot shows the SQL Developer interface with a query window titled 'SF_PROJECT.ITR_RDS_LANDING'. The query contains the following SQL code:

```
--Creation of Schemas
create schema itr_rds;
create schema itr_rds_landing;
create schema itr_dis;

show schemas;
```

The 'Results' tab displays a table with 5 rows and 6 columns: created_on, name, is_default, is_current, and database_name. The data shows the creation of INFORMATION_SCHEMA, ITR_DIS, ITR_RDS, ITR_RDS_LANDING, and PUBLIC schemas.

	created_on	name	is_default	is_current	database_name
1	2022-03-24 10:56:17.600 -0700	INFORMATION_SCHEMA	N	N	SF_PROJECT
2	2022-03-24 10:54:08.682 -0700	ITR_DIS	N	N	SF_PROJECT
3	2022-03-24 10:54:08.232 -0700	ITR_RDS	N	N	SF_PROJECT
4	2022-03-24 10:54:08.503 -0700	ITR_RDS_LANDING	N	Y	SF_PROJECT
5	2022-03-24 10:49:35.777 -0700	PUBLIC	N	N	SF_PROJECT

Query Details: Query duration 41ms, Rows 5.

Fig: 02

3. Created a table named **LOAN** as per the data set.

The screenshot shows the SQL Developer interface with a query window titled 'SF_PROJECT.ITR_RDS_LANDING'. The query contains the following SQL code:

```
create table loan;
desc table loan;
```

The 'Results' tab displays a table with 11 rows and 7 columns: name, type, kind, null?, default, and primary key. The data shows the structure of the LOAN table.

	name	type	kind	null?	default	primary key
1	ID	NUMBER(38,0)	COLUMN	Y	null	N
2	YEAR	NUMBER(38,0)	COLUMN	Y	null	N
3	LOAN_LIMIT	VARCHAR(16777216)	COLUMN	Y	null	N
4	GENDER	VARCHAR(16777216)	COLUMN	Y	null	N
5	APPROV_IN_ADV	VARCHAR(16777216)	COLUMN	Y	null	N
6	LOAN_PURPOSE	VARCHAR(16777216)	COLUMN	Y	null	N
7	CREDIT_WORTHINESS	VARCHAR(16777216)	COLUMN	Y	null	N
8	OPEN_CREDIT	VARCHAR(16777216)	COLUMN	Y	null	N
9	BUSINESS_OR_COMMERCIAL	VARCHAR(16777216)	COLUMN	Y	null	N
10	LOAN_AMOUNT	NUMBER(38,0)	COLUMN	Y	null	N
11	RATE_OF_INTEREST	NUMBER(38,0)	COLUMN	Y	null	N

Query Details: Query duration 72ms, Rows 30.

Fig: 03

~ 3 ~

4. Created Integration object named **s3_int_object**.

The screenshot displays the Snowflake SQL Editor interface. The top section shows a SQL query to create or replace an S3 integration named `s3_int_obj`. The query is as follows:

```
SF_PROJECT.ITR_RDS_LANDING >  
  
61  
62 create or replace storage integration s3_int_obj  
63   type = external_stage  
64   storage_provider = s3  
65   enabled = true  
66   storage_aws_role_arn = 'arn:aws:iam::874545814278:role/flatbucket5_policy_role'  
67   storage_allowed_locations = ('s3://flatbucket5/');  
68  
69 desc integration s3_int_obj;
```

Below the query, the interface shows the 'Results' tab with a table of properties for the integration `s3_int_obj`.

	property	property_type	property_value
1	ENABLED	Boolean	true
2	STORAGE_PROVIDER	String	S3
3	STORAGE_ALLOWED_LOCATIONS	List	s3://flatbucket5/
4	STORAGE_BLOCKED_LOCATIONS	List	
5	STORAGE_AWS_IAM_USER_ARN	String	arn:aws:iam:122191154513:user/d8ia-s-insa5128
6	STORAGE_AWS_ROLE_ARN	String	arn:aws:iam:874545814278:role/flatbucket5_policy_role
7	STORAGE_AWS_EXTERNAL_ID	String	NV27967_SFCSRole=3_WuSLiXwxHihwHITt5JCy2k20AMc=
8	COMMENT	String	

On the right side, the 'Query Details' panel shows the query duration as 190ms and 8 rows returned. Below this, there are two progress bars: one for 'property' (100% filled) and one for 'property_type' (50% filled).

Fig: 04

5. Created external stage named **MY_EXT_STAGE** for loading data structures

The screenshot displays the Snowflake SQL Editor interface. The top section shows a SQL query to create or replace an external stage named `my_ext_stage` in the `sf_project.itr_rds_landing` schema. The query is as follows:

```
SF_PROJECT.ITR_RDS_LANDING >  
  
71  
72  
73 create or replace stage sf_project.itr_rds_landing.my_ext_stage  
74   storage_integration = s3_int_obj  
75   url = 's3://flatbucket5/'  
76   file_format = (type = csv field_delimiter=',' skip_header = 1 null_if = ('NULL','null') empty_field_as_null = true  
77   field_optionally_enclosed_by='');  
78 list @sf_project.itr_rds_landing.my_ext_stage;
```

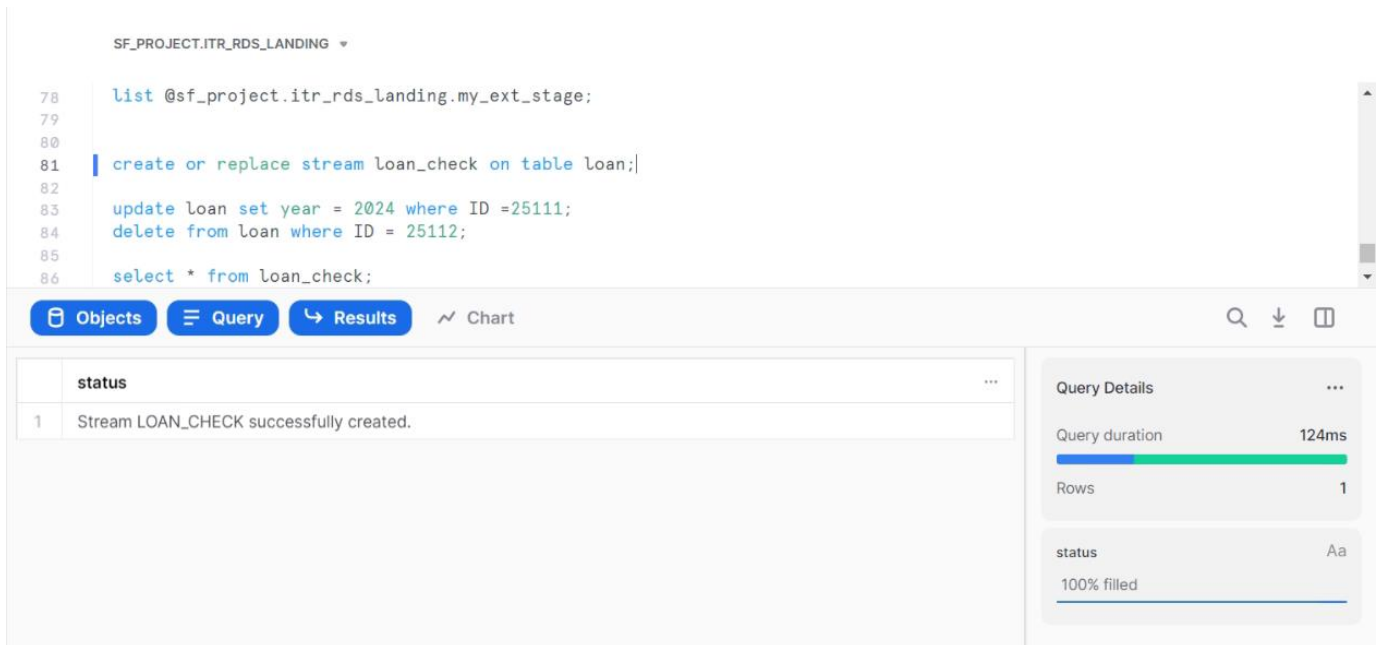
Below the query, the interface shows the 'Results' tab with a table of files in the external stage.

	name	size	md5	last_modified
1	s3://flatbucket5/Loan30.csv	5,168	7909a975fc38d084e2d4ec9265d9f04b	Wed, 23 Mar 2022 11:31:4
2	s3://flatbucket5/Loan8.csv	1,644	ac0513aa3da8b4c7295a61d8a9fc3467	Wed, 23 Mar 2022 11:47:19

On the right side, the 'Query Details' panel shows the query duration as 2.2s and 2 rows returned. Below this, there are two progress bars: one for 'name' (100% filled) and one for 'size' (100% filled).

Fig: 05

6. Created stream **LOAN_CHECK** on table **LOAN**.



```
SF_PROJECT.ITR_RDS_LANDING ▾  
78 list @sf_project.itr_rds_landing.my_ext_stage;  
79  
80  
81 create or replace stream loan_check on table loan;  
82  
83 update loan set year = 2024 where ID =25111;  
84 delete from loan where ID = 25112;  
85  
86 select * from loan_check;
```

Objects Query Results Chart

	status
1	Stream LOAN_CHECK successfully created.

Query Details

Query duration 124ms

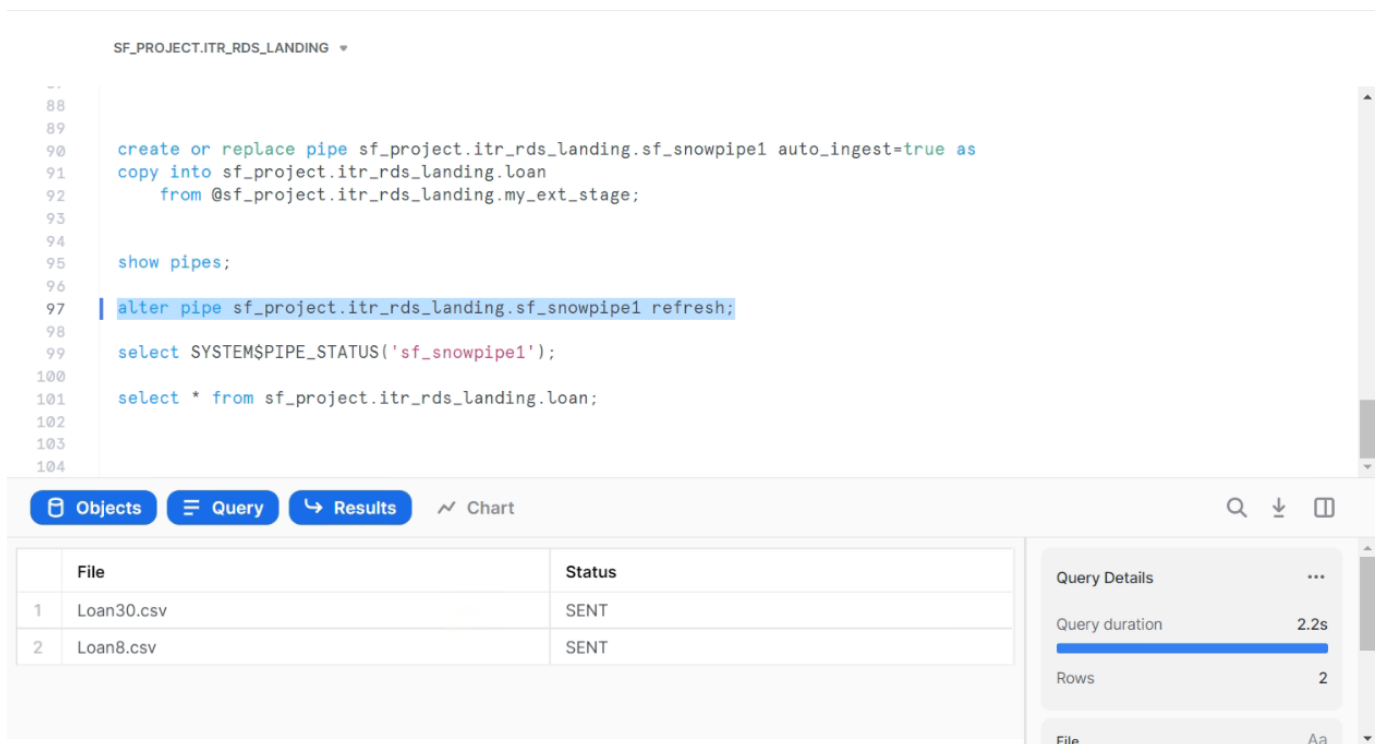
Rows 1

status Aa

100% filled

Fig: 06

7. Created a snowpipe named **SF_SNOWPIPE1** for autoingesting the data from S3 bucket – flatbucket5.



```
SF_PROJECT.ITR_RDS_LANDING ▾  
--  
88  
89  
90 create or replace pipe sf_project.itr_rds_landing.sf_snowpipe1 auto_ingest=true as  
91 copy into sf_project.itr_rds_landing.loan  
92 from @sf_project.itr_rds_landing.my_ext_stage;  
93  
94  
95 show pipes;  
96  
97 alter pipe sf_project.itr_rds_landing.sf_snowpipe1 refresh;  
98  
99 select SYSTEM$PIPE_STATUS('sf_snowpipe1');  
100  
101 select * from sf_project.itr_rds_landing.loan;  
102  
103  
104
```

Objects Query Results Chart

	File	Status
1	Loan30.csv	SENT
2	Loan8.csv	SENT

Query Details

Query duration 2.2s

Rows 2

File Aa

Fig: 07

~ 5 ~

8. Performed SCD operations on consumer table LOAN_TARGET as per changes that happen in the source table LOAN, Task creation and Merge.

```
SF_PROJECT.ITR_RDS_LANDING
143 CREATE TASK loan_task
144     WAREHOUSE = my_compute_warehouse
145     SCHEDULE = '1 minute'
146     WHEN
147         SYSTEM$STREAM_HAS_DATA('loan_check')
148     AS
149     merge into loan_target t
150     using loan_check s
151     on t.id=s.id and (metadata$action='DELETE')
152     when matched and metadata$update='FALSE' then update set rec_version=9999, stream_type='DELETE'
153     when matched and metadata$update='TRUE' then update set rec_version=rec_version-1
154     when not matched then insert (ID ,year,loan_limit ,Gender ,approv_in_adv ,loan_purpose ,Credit_Worthiness
    ,open_credit ,business_or_commercial ,loan_amount ,rate_of_interest ,Interest_rate_spread,Upfront_charges,term
    ,Nee_ammortization ,interest_onlv,Lump_sum_payment,property_value ,Secured_by ,total_units_income ,credit_type
```

status

1	Task LOAN_TASK successfully created.
---	--------------------------------------

Query Details

Query duration 53ms

Rows 1

status 100% filled

Fig: 08-a

```
SF_PROJECT.ITR_RDS_LANDING
155
156
157 ALTER TASK loan_task RESUME;
158
159 select * from loan_target;
160 select * from loan;
161
162 select ID, YEAR, stream_type, rec_version, REC_DATE from loan_target;
```

	ID	YEAR	STREAM_TYPE	REC_VERSION	REC_DATE
1	25111	2,022	INSERT	0	2022-03-24 11:42:31.008 -0700
2	25111	2,020	INSERT	-1	2022-03-24 11:40:23.284 -0700
3	25111	2,019	INSERT	-2	2022-03-24 11:35:21.911 -0700
4	25112	2,019	DELETE	9,999	2022-03-24 11:35:21.911 -0700
5	25113	2,019	INSERT	0	2022-03-24 11:35:21.911 -0700
6	25114	2,019	INSERT	0	2022-03-24 11:35:21.911 -0700
7	25115	2,019	INSERT	0	2022-03-24 11:35:21.911 -0700

Query Details

Query duration 25ms

Rows 40

ID 123

Fig: 08-b

For training purpose, we scheduled it at 1 min. schedule. But to schedule it everyday at 12AM we can use Cronjob.

DATA ANALYSIS ON THE GIVEN DATASET

01. Calculate the total loan amount for gender = 'female' and loan_limit='cf'.

select sum(loan_amount) from loan where gender='Female' and loan_limit='cf';

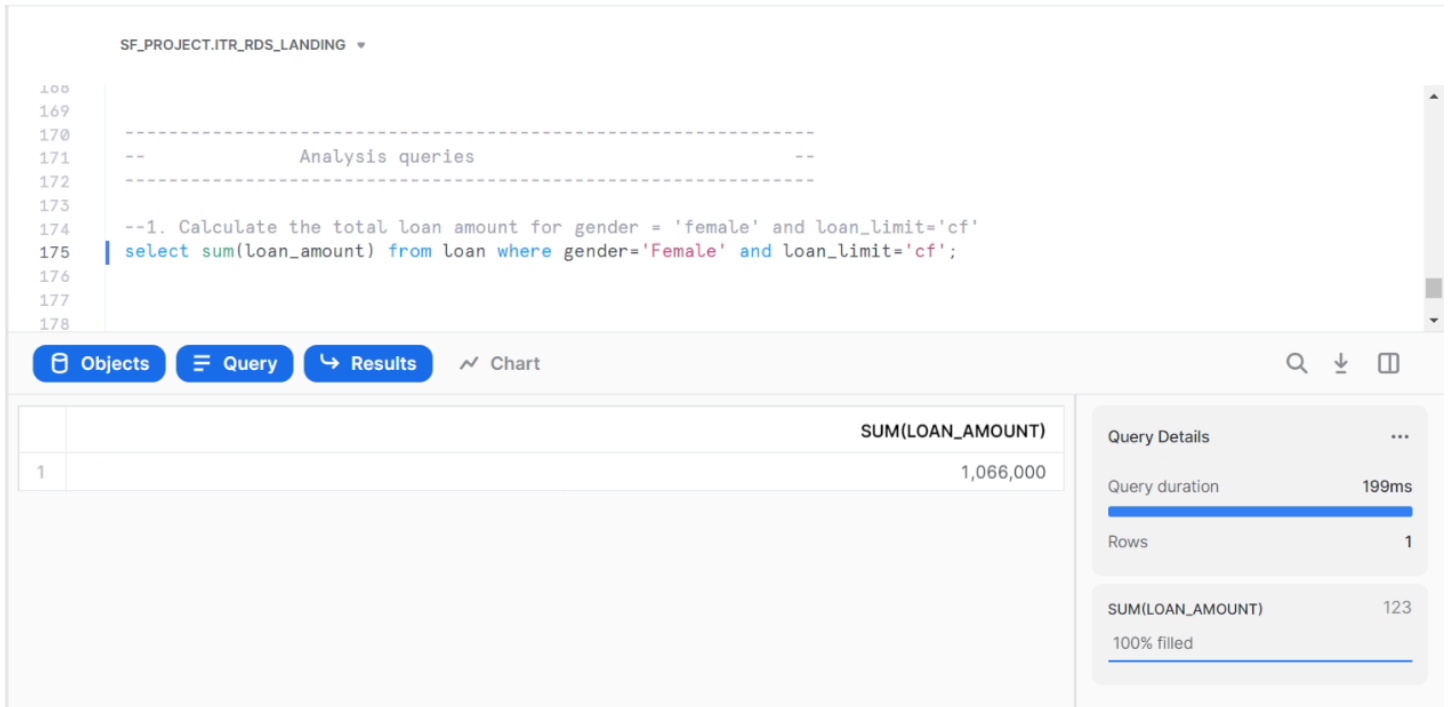


Fig: Sol-01

02. What is the difference in percentage for the number of loan between different valid genders?

select count(id),gender from loan where gender in ('Male','Female') group by gender;

select count() from loan;*

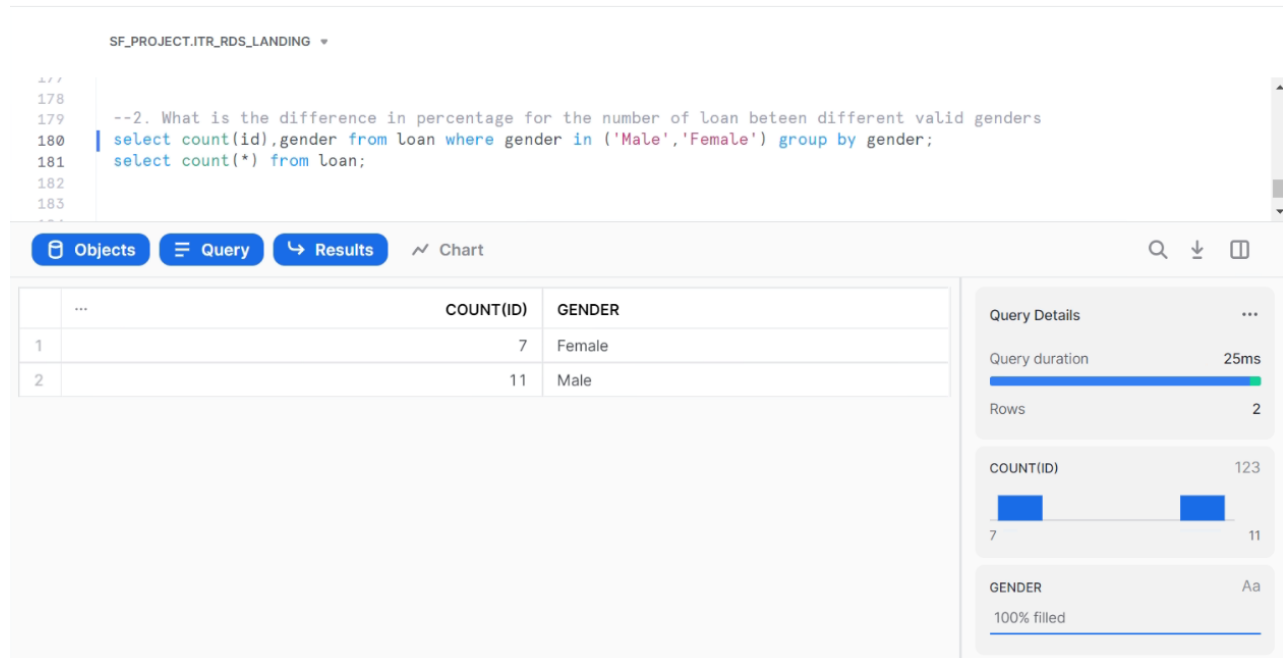


Fig: Sol-02

03. What is the difference in percentage of approve in advance between business and commercial loan?

Select count(loan_amount),business_or_commercial from loan where loan_amount is not null group by business_or_commercial ;

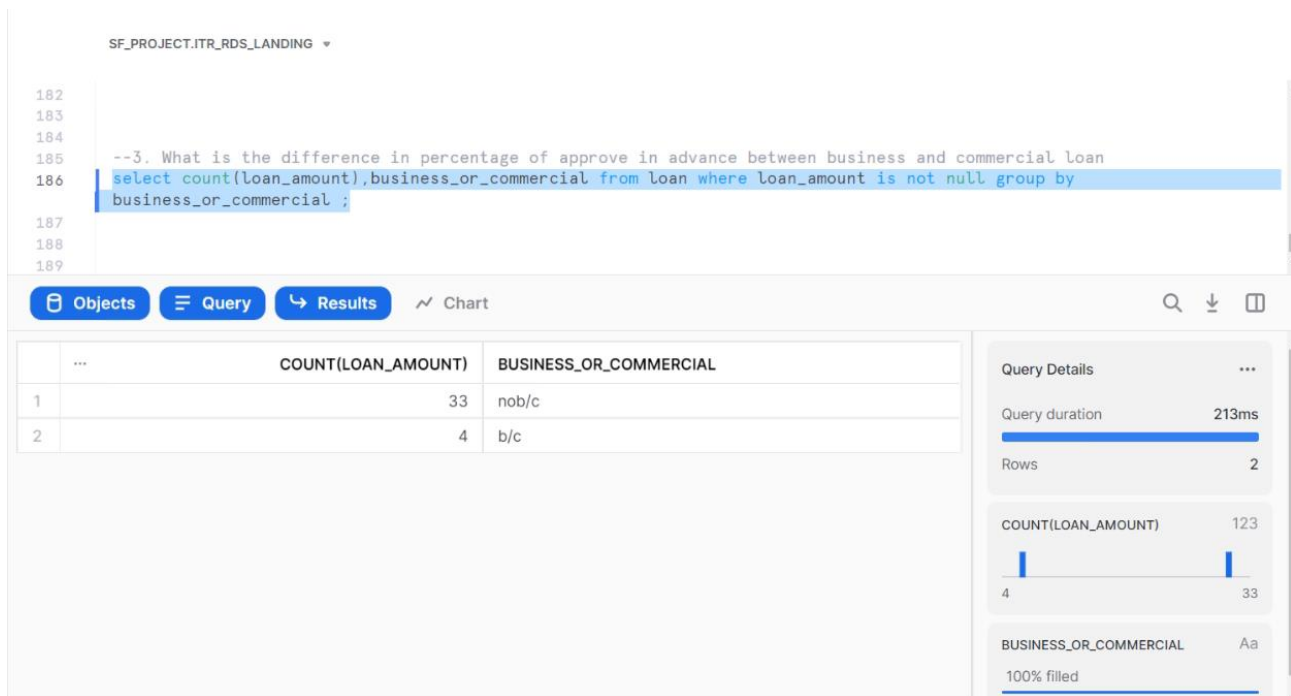


Fig: Sol-03

04. Is there any lumpsum pay for business loan?
select distinct lump_sum_payment from loan ;

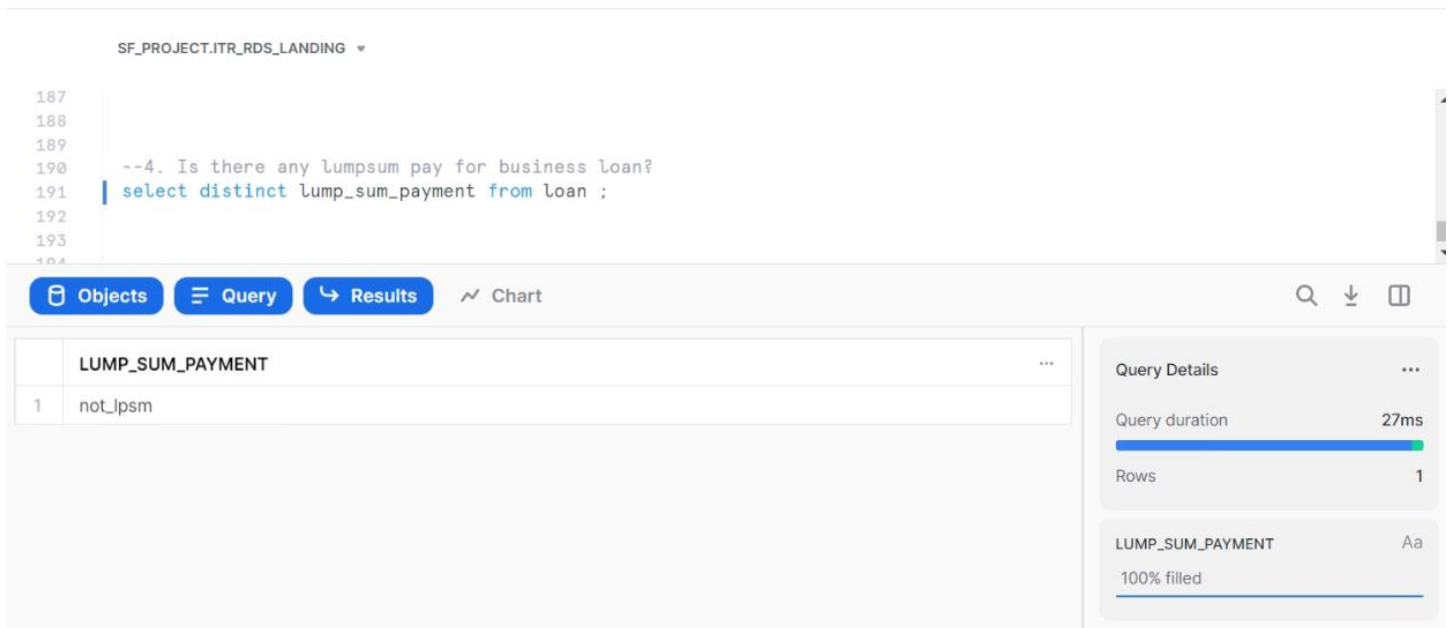


Fig: Sol-04

05. Average credit score for various age group.
select avg(credit_score),age from loan group by age;

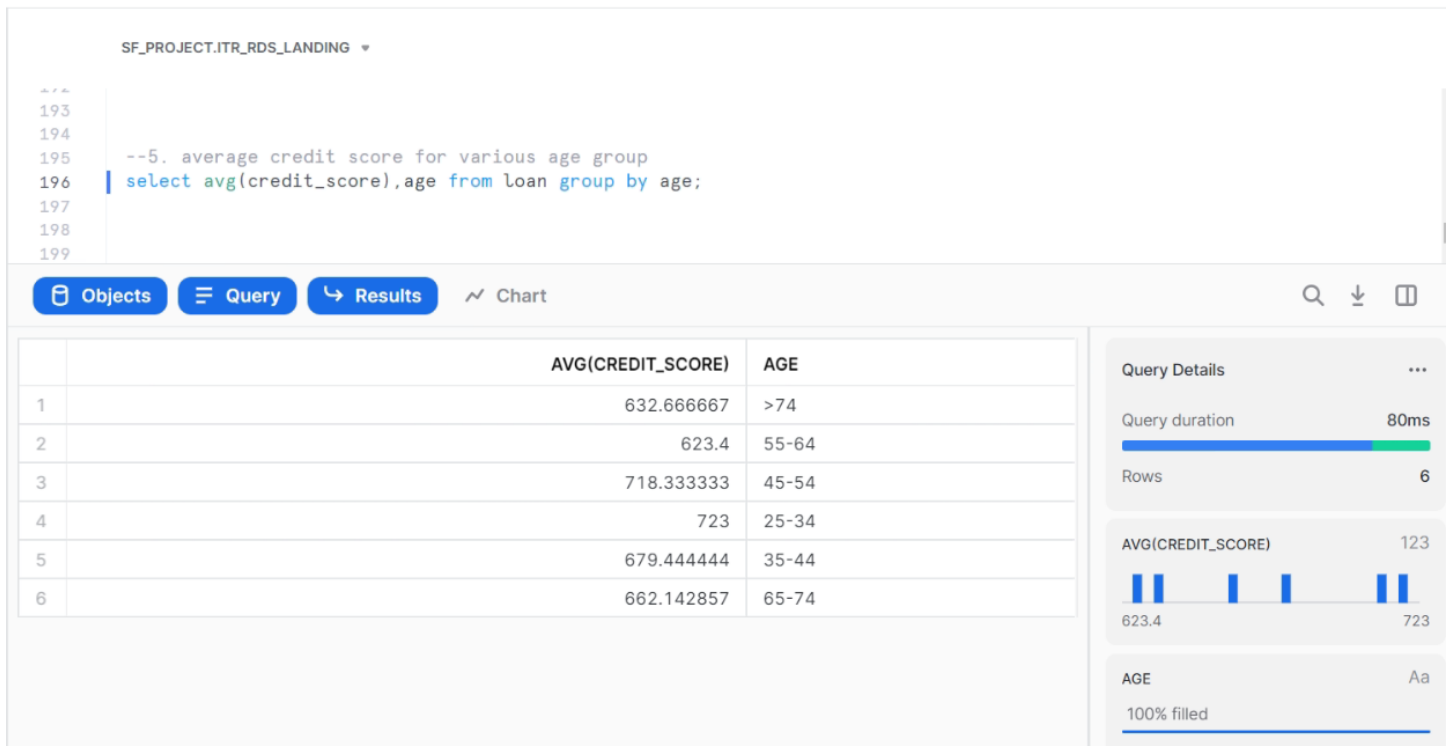


Fig: Sol-05