Assignment 4

**Create database**

Create or replace database demo\_db;

**Create Schema**

create or replace schema demo\_db.Assignment\_4;

**create File Format**

create or replace file format DEMO\_DB.PUBLIC.my\_file\_format

type = 'JSON'

compression = 'AUTO'

STRIP\_OUTER\_ARRAY = True;

**create Info Table**

CREATE OR REPLACE TABLE demo\_db.Assignment\_4.info\_table (

username varchar

,followers numeric

,country varchar

,joined date

,location varchar

,name varchar

,player\_id varchar

,status varchar

,title varchar

);

**Create List Table**

CREATE or replace TABLE list\_table(

username VARCHAR

,is\_live boolean);

**Create Stats Table**

CREATE or replace TABLE stats\_table(

last\_blitz numeric

draw\_blitz numeric

loss\_blitz numeric

win\_blitz: numeric

last\_bullet numeric

draw\_bullet numeric

loss\_bullet numeric

win\_bullet numeric

last\_rapid numeric

draw\_rapid numeric

loss\_rapid numeric

win\_rapid numeric

FIDE numeric );

**Create Stage**

CREATE OR REPLACE STAGE DEMO\_DB.Assignment\_4.MY\_S3\_STAGE

URL='s3://sami-s3-bucket/Data/'

CREDENTIALS=(AWS\_KEY\_ID='AKIA6XXXXXXXXBRN2' AWS\_SECRET\_KEY='4gYj4mT2FozwYwUDXXXXXXXXXX5iL1Tl5/6')

file\_format = my\_file\_format;

list @DEMO\_DB.Assignment\_4.MY\_S3\_STAGE;

**Create pipe for Info Table**

CREATE OR REPLACE PIPE DEMO\_DB.Assignment\_4.info\_table\_pipe

AUTO\_INGEST = TRUE AS

COPY INTO info\_table(username, followers, country, joined, location, name, player\_id, status, title)

FROM (

SELECT $1:username::varchar as username,

$1:followers::numeric as followers,

$1:country::varchar as country,

$1:joined::date as joined,

$1:location::varchar as location,

$1:name::varchar as name,

$1:player\_id::varchar as player\_id,

$1:status::varchar as status,

$1:title::varchar as title

FROM @DEMO\_DB.Assignment\_4.MY\_S3\_STAGE/Info\_file.json

)

FILE\_FORMAT = (FORMAT\_NAME = 'DEMO\_DB.PUBLIC.my\_file\_format')

ON\_ERROR = 'CONTINUE';

alter pipe info\_table\_pipe refresh;

**create pipe for List Table**

CREATE OR REPLACE PIPE DEMO\_DB.Assignment\_4.list\_table\_pipe

AUTO\_INGEST = TRUE AS

COPY INTO list\_table(username,is\_live)

FROM (

SELECT $1:username::varchar as username,

$1:is\_live::boolean as is\_live,

FROM @DEMO\_DB.Assignment\_4.MY\_S3\_STAGE/list\_file.json

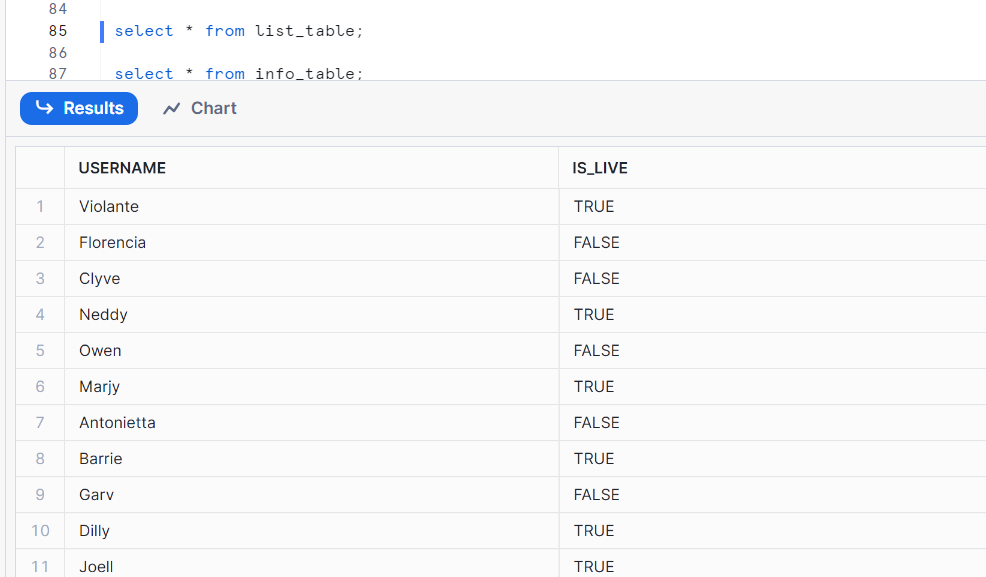
)

FILE\_FORMAT = (FORMAT\_NAME = 'DEMO\_DB.PUBLIC.my\_file\_format')

ON\_ERROR = 'CONTINUE';

alter pipe list\_table\_pipe refresh;

select \* from list\_table;



select \* from info\_table;

A screenshot of a computer

Description automatically generated

**Create pipe Stats\_table\_pipe**

CREATE OR REPLACE PIPE DEMO\_DB.Assignment\_4.stats\_table\_pipe

AUTO\_INGEST = TRUE AS

COPY INTO stats\_table(last\_blitz,draw\_blitz,loss\_blitz,win\_blitz,last\_bullet,draw\_bullet,loss\_bullet,win\_bullet,last\_rapid,draw\_rapid,

loss\_rapid,win\_rapid,FIDE,primary\_key)

FROM (

SELECT

$1:last\_blitz::numeric as last\_blitz,

$1:draw\_blitz::numeric as draw\_blitz,

$1:loss\_blitz::numeric as loss\_blitz,

$1:win\_blitz::numeric as win\_blitz,

$1:last\_bullet::numeric as last\_bullet,

$1:draw\_bullet::numeric as draw\_bullet,

$1:loss\_bullet::numeric as loss\_bullet,

$1:win\_bullet::numeric as win\_bullet,

$1:last\_rapid::numeric as last\_rapid,

$1:draw\_rapid::numeric as draw\_rapid,

$1:loss\_rapidd::numeric as loss\_rapid,

$1:win\_rapid::numeric as win\_rapid,

$1:FIDE::numeric as FIDE,

$1:primary\_key::numeric as primary\_key

FROM @DEMO\_DB.Assignment\_4.MY\_S3\_STAGE/stats\_table.json

)

FILE\_FORMAT = (FORMAT\_NAME = 'DEMO\_DB.PUBLIC.my\_file\_format')

ON\_ERROR = 'CONTINUE';

alter pipe DEMO\_DB.Assignment\_4.stats\_table\_pipe refresh;

select \* from stats\_table;

A screenshot of a computer

Description automatically generated

---Username of the best player by category (blitz, chess, bullet

SELECT i.username

FROM demo\_db.Assignment\_4.info\_table i

JOIN demo\_db.Assignment\_4.stats\_table s ON i.primary\_key = s.primary\_key

ORDER BY s.last\_blitz DESC

LIMIT 1;

SELECT i.username

FROM demo\_db.Assignment\_4.info\_table i

JOIN demo\_db.Assignment\_4.stats\_table s ON i.primary\_key = s.primary\_key

ORDER BY s.last\_bullet DESC

LIMIT 1;

---Full name (or username if null) of the best player and his FIDE elo

SELECT

COALESCE(i.name, i.username) AS best\_player\_name,

s.FIDE AS fide\_elo

FROM demo\_db.Assignment\_4.info\_table i

JOIN demo\_db.Assignment\_4.stats\_table s ON i.primary\_key = s.primary\_key

ORDER BY s.FIDE DESC

LIMIT 1;

---Average elo of premium, staff and basic players

SELECT

i.status,

AVG(s.FIDE) AS average\_elo

FROM demo\_db.Assignment\_4.info\_table i

JOIN demo\_db.Assignment\_4.stats\_table s ON i.primary\_key = s.primary\_key

WHERE i.status IN ('premium', 'staff', 'basic')

GROUP BY i.status;

---Number of professional players and their elo

SELECT

COUNT(\*) AS number\_of\_professionals,

AVG(s.FIDE) AS average\_elo

FROM demo\_db.Assignment\_4.info\_table i

JOIN demo\_db.Assignment\_4.stats\_table s ON i.primary\_key = s.primary\_key

WHERE i.title IS NOT NULL;

---Average FIDE elo by their professional FIDE elo

SELECT

i.title,

AVG(s.FIDE) AS average\_fide\_elo

FROM demo\_db.Assignment\_4.info\_table i

JOIN demo\_db.Assignment\_4.stats\_table s ON i.primary\_key = s.primary\_key

GROUP BY i.title;

---Best player currently on live

SELECT

COALESCE(i.name, i.username) AS best\_live\_player\_name,

s.FIDE AS fide\_elo

FROM demo\_db.Assignment\_4.info\_table i

JOIN demo\_db.Assignment\_4.stats\_table s ON i.primary\_key = s.primary\_key

JOIN demo\_db.Assignment\_4.list\_table l ON i.username = l.username

WHERE l.is\_live = TRUE

ORDER BY s.FIDE DESC

LIMIT 1;