// Set up Environment

-- Create DB, Schema, Warehouse

USE ROLE SYSADMIN;

USE DATABASE USECASE\_DB;

--CREATE DATABASE IF NOT EXISTS USECASE\_DB;

CREATE OR REPLACE SCHEMA TAG\_BASED\_MASKING\_DEMO;

CREATE WAREHOUSE IF NOT EXISTS TAG\_BASED\_MASKING\_DEMO WAREHOUSE\_SIZE=small INITIALLY\_SUSPENDED=true;

-- Create Roles

USE ROLE USERADMIN;

CREATE ROLE IF NOT EXISTS DEVELOPER;

CREATE ROLE IF NOT EXISTS ANALYST;

CREATE ROLE IF NOT EXISTS SUPPORT;

CREATE ROLE IF NOT EXISTS REPORTING;

CREATE ROLE IF NOT EXISTS TAG\_ADMIN;

-- Grants for hierarchy

GRANT ROLE DEVELOPER TO ROLE SYSADMIN;

GRANT ROLE ANALYST TO ROLE SYSADMIN;

GRANT ROLE SUPPORT TO ROLE SYSADMIN;

GRANT ROLE REPORTING TO ROLE SYSADMIN;

GRANT ROLE TAG\_ADMIN TO ROLE SYSADMIN;

-- Grant usage

USE ROLE SYSADMIN;

GRANT USAGE ON DATABASE USECASE\_DB TO ROLE DEVELOPER;

GRANT USAGE ON DATABASE USECASE\_DB TO ROLE ANALYST;

GRANT USAGE ON DATABASE USECASE\_DB TO ROLE SUPPORT;

GRANT USAGE ON DATABASE USECASE\_DB TO ROLE REPORTING;

GRANT USAGE ON DATABASE USECASE\_DB TO ROLE TAG\_ADMIN;

GRANT USAGE ON SCHEMA USECASE\_DB.TAG\_BASED\_MASKING\_DEMO TO ROLE DEVELOPER;

GRANT USAGE ON SCHEMA USECASE\_DB.TAG\_BASED\_MASKING\_DEMO TO ROLE ANALYST;

GRANT USAGE ON SCHEMA USECASE\_DB.TAG\_BASED\_MASKING\_DEMO TO ROLE SUPPORT;

GRANT USAGE ON SCHEMA USECASE\_DB.TAG\_BASED\_MASKING\_DEMO TO ROLE REPORTING;

GRANT USAGE ON SCHEMA USECASE\_DB.TAG\_BASED\_MASKING\_DEMO TO ROLE TAG\_ADMIN;

-- Not using recommended functional/access roles strategy for demo purposes

-- Grant privileges

GRANT SELECT, INSERT, UPDATE, DELETE ON ALL TABLES IN SCHEMA USECASE\_DB.TAG\_BASED\_MASKING\_DEMO TO ROLE DEVELOPER;

GRANT SELECT ON ALL TABLES IN SCHEMA USECASE\_DB.TAG\_BASED\_MASKING\_DEMO TO ROLE ANALYST;

GRANT SELECT ON ALL TABLES IN SCHEMA USECASE\_DB.TAG\_BASED\_MASKING\_DEMO TO ROLE REPORTING;

GRANT ALL ON ALL TABLES IN SCHEMA USECASE\_DB.TAG\_BASED\_MASKING\_DEMO TO ROLE SUPPORT;

GRANT USAGE ON WAREHOUSE TAG\_BASED\_MASKING\_DEMO TO ROLE DEVELOPER;

GRANT USAGE ON WAREHOUSE TAG\_BASED\_MASKING\_DEMO TO ROLE ANALYST;

GRANT USAGE ON WAREHOUSE TAG\_BASED\_MASKING\_DEMO TO ROLE SUPPORT;

GRANT USAGE ON WAREHOUSE TAG\_BASED\_MASKING\_DEMO TO ROLE REPORTING;

-- Future grants

USE ROLE SECURITYADMIN;

GRANT SELECT, INSERT, UPDATE, DELETE ON FUTURE TABLES IN SCHEMA USECASE\_DB.TAG\_BASED\_MASKING\_DEMO TO ROLE DEVELOPER;

GRANT SELECT ON FUTURE TABLES IN SCHEMA USECASE\_DB.TAG\_BASED\_MASKING\_DEMO TO ROLE ANALYST;

GRANT SELECT ON FUTURE TABLES IN SCHEMA USECASE\_DB.TAG\_BASED\_MASKING\_DEMO TO ROLE REPORTING;

GRANT ALL ON FUTURE TABLES IN SCHEMA USECASE\_DB.TAG\_BASED\_MASKING\_DEMO TO ROLE SUPPORT;

GRANT CREATE MASKING POLICY ON SCHEMA USECASE\_DB.TAG\_BASED\_MASKING\_DEMO TO ROLE TAG\_ADMIN;

//Tag Grants

-- Grants to create and apply tags + apply masking policy to TAG\_ADMIN role

USE ROLE ACCOUNTADMIN;

GRANT APPLY TAG ON ACCOUNT TO ROLE TAG\_ADMIN;

GRANT CREATE TAG ON SCHEMA USECASE\_DB.TAG\_BASED\_MASKING\_DEMO TO ROLE TAG\_ADMIN;

GRANT APPLY MASKING POLICY ON ACCOUNT TO ROLE TAG\_ADMIN; // can use SECURITYADMIN role as well

//Crete Tag

-- Create tag

USE ROLE TAG\_ADMIN;

USE SCHEMA USECASE\_DB.TAG\_BASED\_MASKING\_DEMO;

CREATE OR REPLACE TAG PII ALLOWED\_VALUES 'PUBLIC','RESTRICTED','CONFIDENTIAL';

//Create Masking Policies

CREATE OR REPLACE MASKING POLICY PII\_DEMO\_STRING AS (val string) returns string ->

CASE

-- Listed from most to least restricted order

WHEN SYSTEM$GET\_TAG\_ON\_CURRENT\_COLUMN('PII') = 'CONFIDENTIAL'

THEN

CASE

WHEN CURRENT\_ROLE() IN ('SUPPORT') THEN VAL

ELSE '\*\*\* PII CONFIDENTIAL - STRING IS MASKED \*\*\*'

END

WHEN SYSTEM$GET\_TAG\_ON\_CURRENT\_COLUMN('PII') = 'RESTRICTED'

THEN

CASE

WHEN CURRENT\_ROLE() IN ('SUPPORT','DEVELOPER')

THEN val

ELSE '\*\*\* PII RESTRICTED - STRING IS MASKED \*\*\*'

END

WHEN SYSTEM$GET\_TAG\_ON\_CURRENT\_COLUMN('PII') = 'PUBLIC'

THEN val

ELSE '\*\*\* PII MASKING FAILED: COLUMN IS NOT MAPPED TO A PII TYPE. \*\*\*'

END;

CREATE OR REPLACE MASKING POLICY PII\_DEMO\_NUMBER AS (val number) returns number ->

CASE

-- Listed from most to least restricted order

WHEN SYSTEM$GET\_TAG\_ON\_CURRENT\_COLUMN('PII') = 'CONFIDENTIAL'

THEN

CASE

WHEN CURRENT\_ROLE() IN ('SUPPORT') THEN VAL

ELSE NULL

END

WHEN SYSTEM$GET\_TAG\_ON\_CURRENT\_COLUMN('PII') = 'RESTRICTED'

THEN

CASE

WHEN CURRENT\_ROLE() IN ('SUPPORT','DEVELOPER')

THEN val

ELSE NULL

END

WHEN SYSTEM$GET\_TAG\_ON\_CURRENT\_COLUMN('PII') = 'PUBLIC'

THEN val

ELSE NULL

END;

CREATE OR REPLACE MASKING POLICY PII\_DEMO\_DATE AS (val date) returns date ->

CASE

-- Listed from most to least restricted order

WHEN SYSTEM$GET\_TAG\_ON\_CURRENT\_COLUMN('PII') = 'CONFIDENTIAL'

THEN

CASE

WHEN CURRENT\_ROLE() IN ('SUPPORT') THEN VAL

ELSE NULL

END

WHEN SYSTEM$GET\_TAG\_ON\_CURRENT\_COLUMN('PII') = 'RESTRICTED'

THEN

CASE

WHEN CURRENT\_ROLE() IN ('SUPPORT','DEVELOPER')

THEN val

ELSE NULL

END

WHEN SYSTEM$GET\_TAG\_ON\_CURRENT\_COLUMN('PII') = 'PUBLIC'

THEN val

ELSE NULL

END;

SHOW MASKING POLICIES;

/\*USE ROLE ACCOUNTADMIN;

DROP MASKING POLICY PII\_DEMO\_STRING;

DROP MASKING POLICY PII\_DEMO\_NUMBER;

DROP MASKING POLICY PII\_DEMO\_DATE;

\*/

//Apply Masking Policies to Tag

-- Apply masking policies to tag

USE ROLE TAG\_ADMIN;

ALTER TAG PII SET

MASKING POLICY PII\_DEMO\_STRING,

MASKING POLICY PII\_DEMO\_NUMBER,

MASKING POLICY PII\_DEMO\_DATE;

SHOW TAGS;

//Create/populate dummy table for testing

-- Create dummy table for testing

USE ROLE SYSADMIN;

USE WAREHOUSE TAG\_BASED\_MASKING\_DEMO;

CREATE OR REPLACE TABLE USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE (ID NUMBER, NAME VARCHAR, DOB DATE, SSN VARCHAR,DEPT VARCHAR);

-- Insert random data into dummy table

INSERT INTO USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE

SELECT UNIFORM(10000,99999,ABS(RANDOM())) AS ID, RANDSTR(50,RANDOM()) AS NAME, '1950-01-01'::DATE AS DOB, UNIFORM(111111111,999999999,ABS(RANDOM())) AS SSN, RANDSTR(10,RANDOM()) AS DEPT

FROM TABLE(GENERATOR(ROWCOUNT => 50));

SELECT \* FROM USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE

ORDER BY ID;

//Apply Tags to Dummy table columns

-- Apply tag to columns

ALTER TABLE USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE MODIFY COLUMN ID SET TAG PII ='RESTRICTED';

ALTER TABLE USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE MODIFY COLUMN NAME SET TAG PII ='PUBLIC';

ALTER TABLE USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE MODIFY COLUMN DOB SET TAG PII ='CONFIDENTIAL';

ALTER TABLE USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE MODIFY COLUMN SSN SET TAG PII ='CONFIDENTIAL';

ALTER TABLE USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE MODIFY COLUMN DEPT SET TAG PII ='RESTRICTED';

DESC TABLE USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE;

//Let's Test

USE ROLE SYSADMIN;

SELECT \* FROM USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE;

USE ROLE DEVELOPER;

SELECT \* FROM USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE;

USE ROLE ANALYST;

SELECT \* FROM USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE;

USE ROLE SUPPORT;

SELECT \* FROM USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE;

USE ROLE REPORTING;

SELECT \* FROM USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE;

//clean up fdfd

USE ROLE SYSADMIN;

ALTER TABLE USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE MODIFY COLUMN ID UNSET TAG PII;

ALTER TABLE USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE MODIFY COLUMN NAME UNSET TAG PII;

ALTER TABLE USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE MODIFY COLUMN DOB UNSET TAG PII;

ALTER TABLE USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE MODIFY COLUMN SSN UNSET TAG PII;

ALTER TABLE USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE MODIFY COLUMN DEPT UNSET TAG PII;

ALTER TAG PII UNSET

MASKING POLICY PII\_DEMO\_STRING,

MASKING POLICY PII\_DEMO\_NUMBER,

MASKING POLICY PII\_DEMO\_DATE;

DROP MASKING POLICY PII\_DEMO\_DATE;

DROP MASKING POLICY PII\_DEMO\_NUMBER;

DROP MASKING POLICY PII\_DEMO\_STRING;

DROP TAG PII;

//Account Usage (has Latency)

--This Account Usage view can be used to identify the associations between objects and tags.

select \* from snowflake.account\_usage.tags

order by tag\_name;

select \* from snowflake.account\_usage.tag\_references

order by tag\_name;

//Information Schema (No Latency)

--Retrieve the list of tags associated with the table EMPLOYEE:

select \*

from table(USECASE\_DB.information\_schema.tag\_references('EMPLOYEE', 'table'));

--Retrieve the list of tags associated on the column EMPLOYEE.SSN:

select \*

from table(USECASE\_DB.information\_schema.tag\_references('EMPLOYEE.SSN', 'COLUMN'));

--Retrieve the list of tags that are assigned to every column in the table EMPLOYEE:

select \*

from table(USECASE\_DB.information\_schema.tag\_references\_all\_columns('EMPLOYEE', 'table'));

-- To return the tag value associated with the specified tag, and the Snowflake object or column, use the SYSTEM$GET\_TAG function.

select system$get\_tag('PII', 'USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE', 'table'); --NULL as no tag is set at table level

select system$get\_tag('PII', 'USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE.SSN', 'column'); --CONFIDENTIAL

select system$get\_tag('PII', 'USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE.ID', 'column'); -- RESTRICTED

// check if the same policy is applied on different tables once tags are set at a table level

USE ROLE SYSADMIN;

USE WAREHOUSE TAG\_BASED\_MASKING\_DEMO;

USE SCHEMA USECASE\_DB.TAG\_BASED\_MASKING\_DEMO;

CREATE OR REPLACE TABLE USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE\_INFO(employee\_id number,employee\_ssn string);

--TRUNCATE TABLE USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE\_INFO;

insert into USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE\_INFO values (1,'111-11-1111');

insert into USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE\_INFO values (2,'222-11-1111');

insert into USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE\_INFO values (3,'333-11-1111');

insert into USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE\_INFO values (4,'444-11-1111');

insert into USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE\_INFO values (5,'555-11-1111');

insert into USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE\_INFO values (6,'666-11-1111');

insert into USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE\_INFO values (7,'777-11-1111');

insert into USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE\_INFO values (8,'888-11-1111');

insert into USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE\_INFO values (9,'999-11-1111');

insert into USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE\_INFO values (10,'000-11-1111');

select \* From USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE\_INFO;

//Apply SSN = Confidential tag

USE ROLE SYSADMIN;

ALTER TABLE USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE\_INFO MODIFY COLUMN EMPLOYEE\_SSN SET TAG PII ='CONFIDENTIAL';

//Let's Test

USE ROLE SYSADMIN;

SELECT \* FROM USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE\_INFO;

USE ROLE DEVELOPER;

SELECT \* FROM USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE\_INFO;

USE ROLE ANALYST;

SELECT \* FROM USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE\_INFO;

USE ROLE SUPPORT;

SELECT \* FROM USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE\_INFO;

USE ROLE REPORTING;

SELECT \* FROM USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE\_INFO;

//Clean up

USE ROLE SYSADMIN;

DROP TABLE USECASE\_DB.TAG\_BASED\_MASKING\_DEMO.EMPLOYEE\_INFO;