

Oracle SQL

Day 4

Day 4

- Recap
- Oracle Dictionary – Explore Metadata
- Constraints
 - Check Constraint
 - Unique Constraint
 - Renaming & Dropping Constraints
 - Enabling & Disabling Constraints
- Indexes
 - Create & Drop Index Statements
 - Indexes & Performance

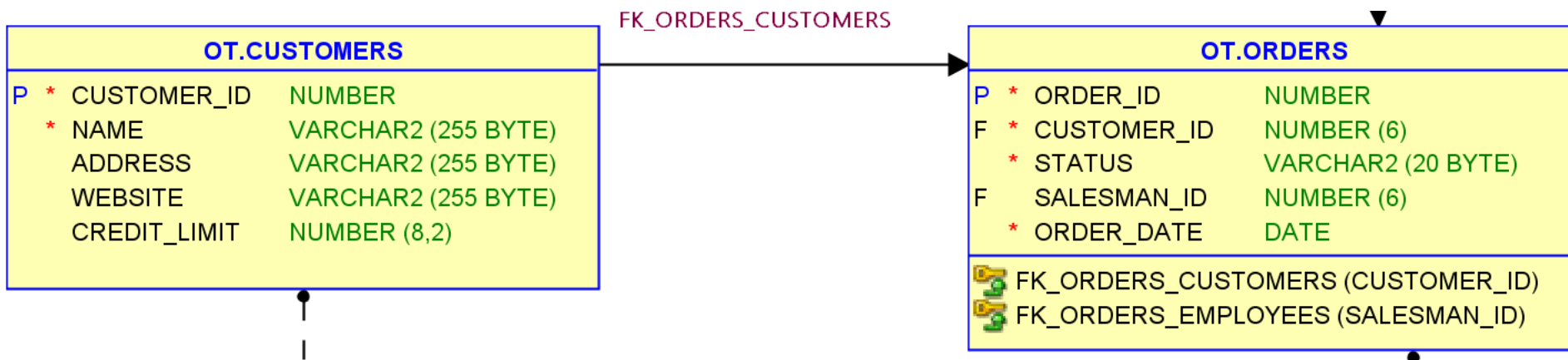
Explore metadata

Finding the relationships in the data dictionary: foreign keys

Example: FK_ORDERS_CUSTOMERS

On: OT.ORDERS.CUSTOMER_ID

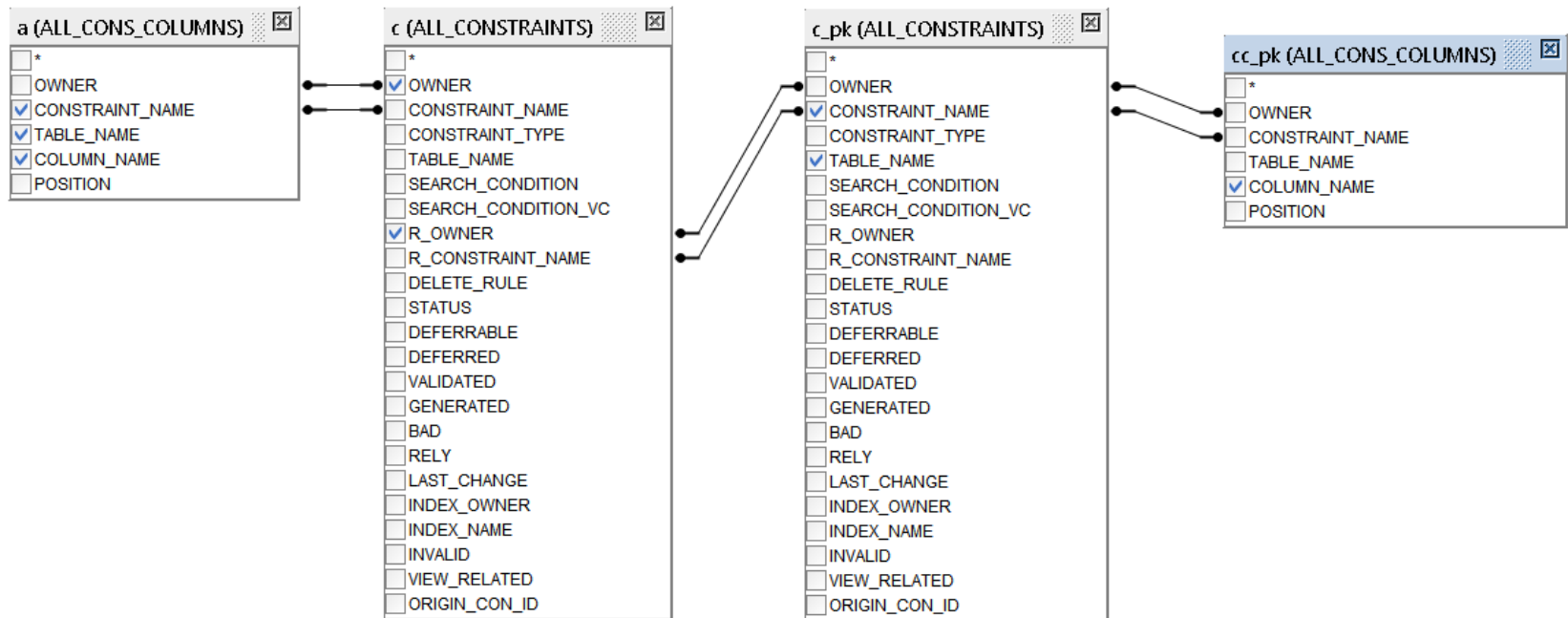
Origin: OT.CUSTOMERS.CUSTOMER_ID



Explore metadata

Finding the relationships in the data dictionary:

Tables: ALL_CONSTRAINTS, ALL_CONS_COLUMNS



Explore metadata

```
SELECT a.table_name
      , a.column_name
      , a.constraint_name
      , c.owner
      , c.r_owner          "Ref Owner"
      , ref_tbl.table_name  "Ref Table"
      , ref_tbl.constraint_name "Ref Constraint"
      , ref_col.column_name "Ref Column"
FROM all_cons_columns a
JOIN all_constraints c ON (a.owner=c.owner
                          AND a.constraint_name=c.constraint_name)
JOIN all_constraints ref_tbl ON (c.r_owner=ref_tbl.owner
                                AND c.r_constraint_name = ref_tbl.constraint_name )
JOIN all_cons_columns ref_col ON (ref_col.owner= ref_tbl.owner
                                AND ref_col.constraint_name = ref_tbl.constraint_name )
WHERE a.owner = 'OT'
      AND c.constraint_type = 'R' -- Foreign Keys
      AND a.table_name IN ( 'ORDERS');
```

TABLE_NAME	COLUMN_NAME	CONSTRAINT_NAME	OWNER	Ref Owner	Ref Table	Ref Constraint	Ref Column
ORDERS	SALESMAN_ID	FK_ORDERS_EMPLOYEES	OT	OT	EMPLOYEES	SYS_C005131	EMPLOYEE_ID
ORDERS	CUSTOMER_ID	FK_ORDERS_CUSTOMERS	OT	OT	CUSTOMERS	SYS_C005143	CUSTOMER_ID

Constraints

Types:

- **NOT NULL** - prohibits a database value from being null.
- **UNIQUE** - prohibits multiple rows from having the same value in the same column or combination of columns but allows some values to be null.
- **PRIMARY KEY** - combines a NOT NULL constraint and a unique constraint in a single declaration. That is, it prohibits multiple rows from having the same value in the same column or combination of columns and prohibits values from being null.
- **FOREIGN KEY** - requires values in one table to match values in another table.
- **CHECK** - requires a value in the database to comply with a specified condition.
- *Others* – not in scope...

Creating Constraints

In CREATE TABLE:

- Can be declared in the column definition, if single column. Multiple column constraints have to be defined at the end of the CREATE TABLE statement.

Or, after table creation, using ALTER TABLE

Creating Constraints

Not NULL:

```
CREATE TABLE name (column datatype NOT NULL ...
```

Unique:

```
CREATE TABLE name (columnlist, CONSTRAINT name UNIQUE  
(columnlist)...
```

Primary Key:

```
CREATE TABLE name (column datatype PRIMARY KEY...
```

or

```
CREATE TABLE name (columnlist, CONSTRAINT name PRIMARY KEY  
(columnlist)...
```

Foreign Key:

```
CREATE TABLE name (columnlist, CONSTRAINT name FOREIGN KEY  
(columnlist) REFERENCES schema.table.column ...
```

Check:

```
CREATE TABLE name (column datatype CONSTRAINT name CHECK  
(condition)...
```


Working with Constraints

Create with ALTER TABLE:

```
ALTER TABLE table_name ADD CONSTRAINT name  
UNIQUE (columnlist);
```

Rename:

```
ALTER TABLE table_name RENAME CONSTRAINT name1  
TO name2;
```

Disable

```
ALTER TABLE table_name DISABLE CONSTRAINT name;
```

Enable:

```
ALTER TABLE table_name ENABLE CONSTRAINT name;
```

Drop:

```
ALTER TABLE table_name DROP CONSTRAINT name;
```

Indexes

Indexes are used to enhance database performance.

When to use:

- Columns used in WHERE and JOIN clauses
- Foreign keys (these are used in JOINS)

Oracle automatically creates an index for PK columns.

Syntax:

```
CREATE [UNIQUE] INDEX name ON tablename (columnlist);
```

```
■
```

```
DROP INDEX [schema_name.] index_name;
```

Unique Indexes

Are really constraints because they are used to enforce integrity rather than for performance reasons.

When the table is populated, creation will fail if there are duplicates on the index key.

ROWNUM

```
select name, credit_limit from ot.customers where
credit_limit in
(SELECT credit_limit FROM
  (
    SELECT distinct credit_limit, ROWNUM AS rn
    FROM
      (
        SELECT distinct credit_limit
        FROM      ot.customers
        ORDER BY   credit_limit DESC
      ) temp
  )
WHERE rn = 1);
```

References

Resource	Location
Oracle Data Dictionary	https://docs.oracle.com/database/121/ADMIN/tables.htm#ADMIN01508
Oracle Data Dictionary detailed	https://docs.oracle.com/database/121/nav/catalog_views.htm
ALL_CONSTRAINTS	https://docs.oracle.com/database/121/REFRN/GUID-9C96DA92-CFE0-4A3F-9061-C5ED17B43EFE.htm#REFRN20047
Constraints script	https://oracle-base.com/dba/script?category=monitoring&file=fks.sql



THANK YOU

average 45%