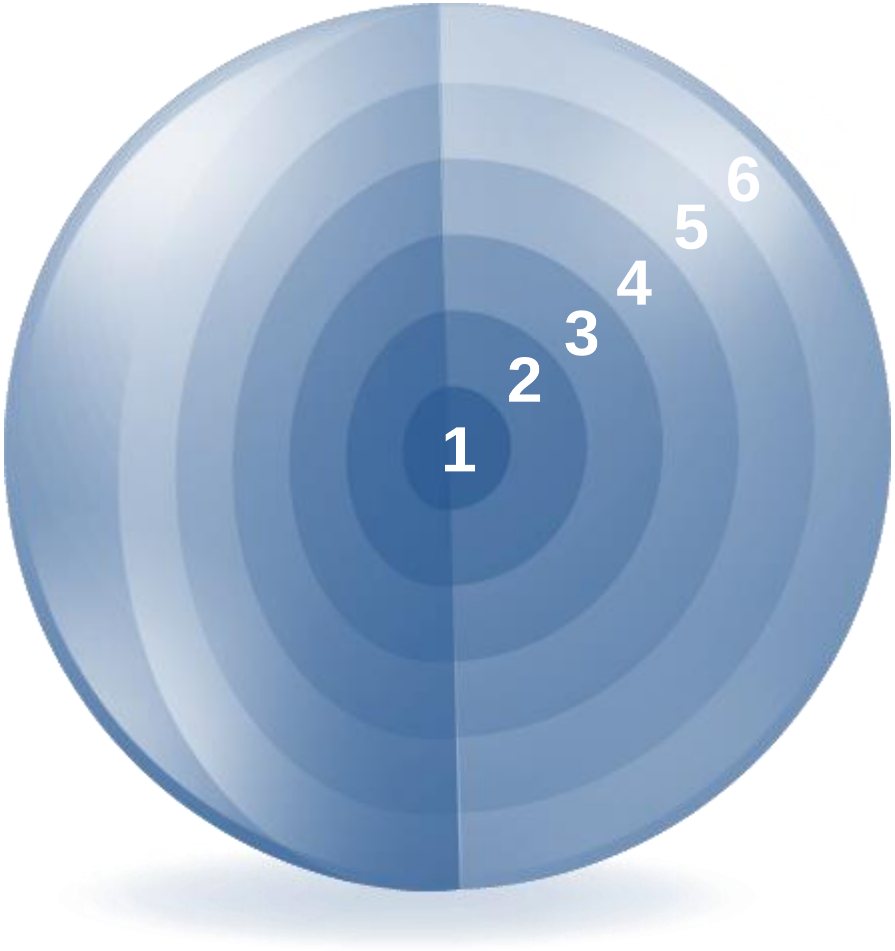
Lesson 10

Using DDL Statements to Create and Manage Tables

ORACLE

Session Summary



1. Categorize the main database objects

2. Review the table structure

3. List the data types that are available for columns

4. Create a simple table

5. Explain how constraints are created at the time of table creation

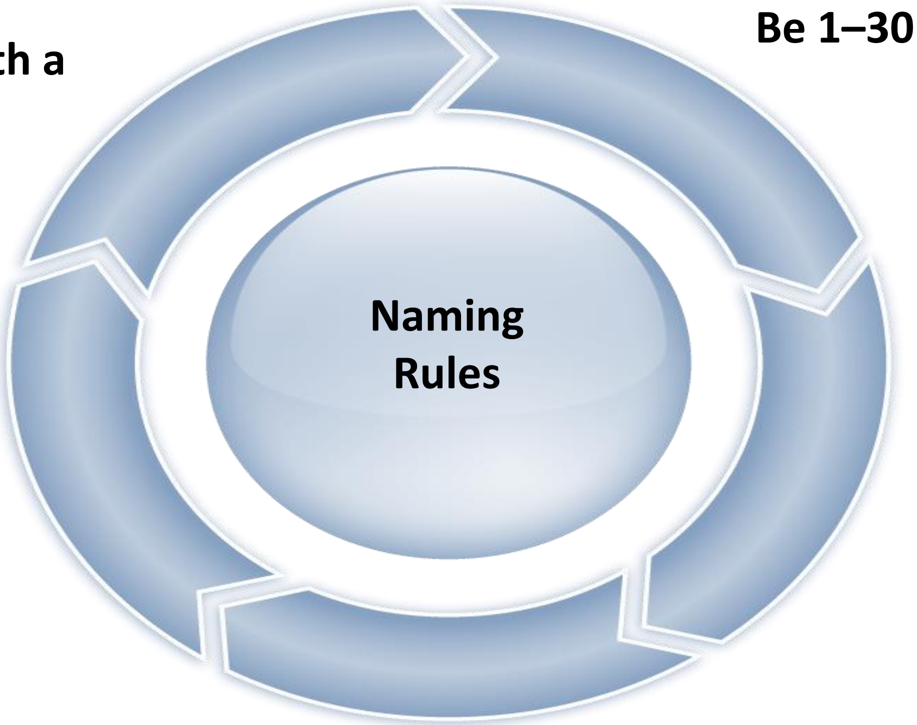
6. Describe how schema objects work

Database Objects

Object	Description
Table	Basic unit of storage; composed of rows
View	Logically represents subsets of data from one or more tables
Sequence	Generates numeric values
Index	Improves the performance of some queries
Synonym	Gives alternative name to an object

Naming Rules

Table names and column names must:



Be 1–30 characters long

Not be an Oracle server–reserved word

Not duplicate the name of another object owned by the same user

Contain only A–Z, a–z, 0–9, _, $, and #

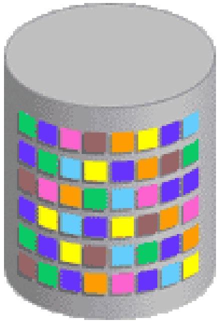
Begin with a letter

CREATE TABLE Statement

- You must have:
  - The CREATE TABLE privilege
  - A storage area

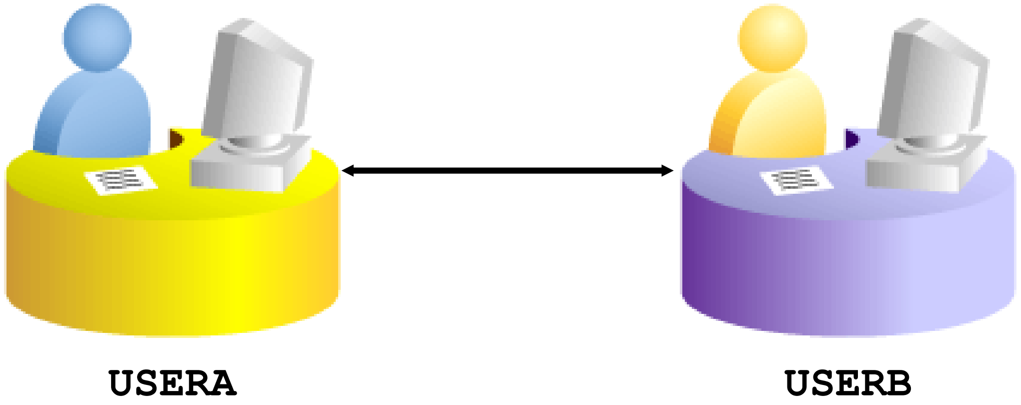
```
CREATE TABLE [schema.]table
          (column datatype [DEFAULT expr][, ...]);
```

- You specify:
  - The table name
  - The column name, column data type, and column size



Referencing Anonther User’s Tables

- Tables belonging to other users are not in the user’s schema.
- You should use the owner’s name as a prefix to those tables.



```
SELECT *
FROM userB.employees;
```

```
SELECT *
FROM userA.employees;
```

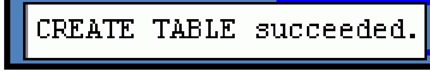
DEFAULT Option

- Specify a default value for a column during an insert.

```
... hire_date DATE DEFAULT SYSDATE, ...
```

- Literal values, expressions, or SQL functions are legal values.
- Another column’s name or a pseudocolumn are illegal values.
- The default data type must match the column data type.

```
CREATE TABLE hire_dates
        (id              NUMBER(8),
         hire_date DATE DEFAULT SYSDATE);
```

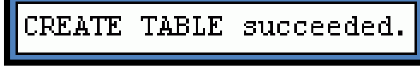


```
CREATE TABLE succeeded.
```

Creating Tables

– Create the table:

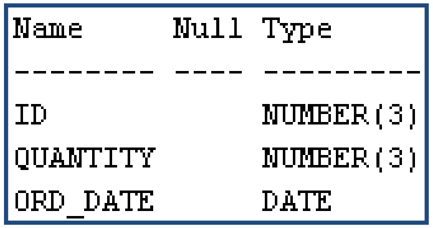
```
CREATE   TABLE  ord
         ( ID  NUMBER (3),
          quantity  NUMBER (3),
          ord_date  DATE  DEFAULT  SYSDATE) ;
```



```
CREATE TABLE succeeded.
```

– Confirm table creation:

```
DESCRIBE   ord ;
```



```
Name     Null Type
-------- ---- ---------
ID            NUMBER(3)
QUANTITY      NUMBER(3)
ORD_DATE      DATE
```

Data Types

Data Type	Description
VARCHAR2(*size*)	Variable-length character data
CHAR(*size*)	Fixed-length character data
NUMBER(*p, s)*	Variable-length numeric data
DATE	Date and time values
LONG	Variable-length character data (up to 2 GB)
CLOB	Character data (up to 4 GB)
RAW and LONG RAW	Raw binary data
BLOB	Binary data (up to 4 GB)
BFILE	Binary data stored in an external file (up to 4 GB)
ROWID	A base-64 number system representing the unique address of a row in its table

Datetime Data Types

You can use several datetime data types:

Data Type	Description
`TIMESTAMP`	Date with fractional seconds
`INTERVAL YEAR TO MONTH`	Stored as an interval of years and months
`INTERVAL DAY TO SECOND`	Stored as an interval of days, hours, minutes, and seconds

Including Constraints

Constraints enforce rules at the table level.

Constraints prevent the deletion of a table if there are dependencies.

The following constraint types are valid:

NOT NULL

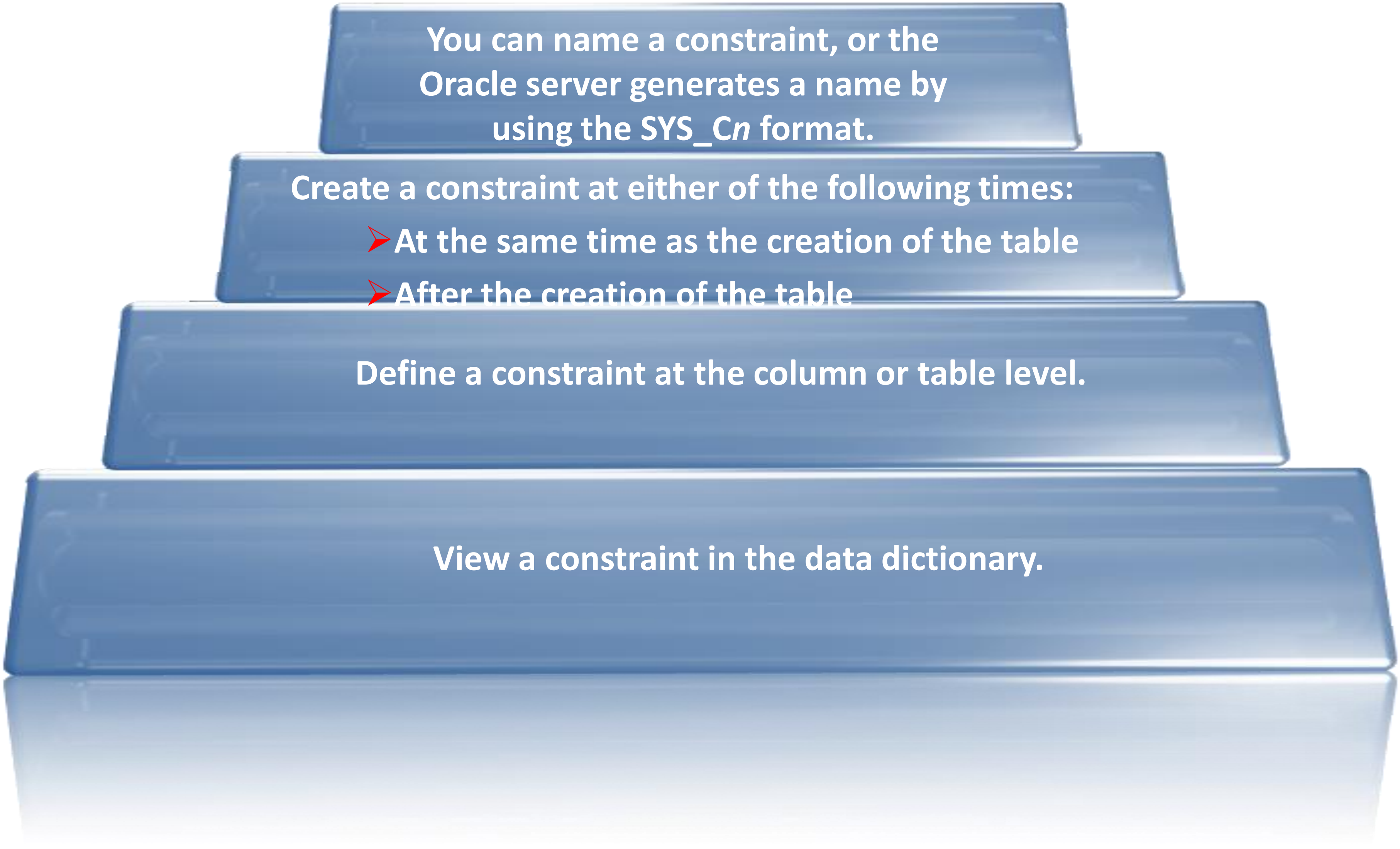
UNIQUE

PRIMARY KEY

FOREIGN KEY

CHECK

Constraint Guidelines



You can name a constraint, or the Oracle server generates a name by using the SYS_C*n* format.

Create a constraint at either of the following times:

- At the same time as the creation of the table
- After the creation of the table

Define a constraint at the column or table level.

View a constraint in the data dictionary.

Defining Constraints

– Syntax:

```
CREATE TABLE [schema.]table
      (column datatype [DEFAULT expr]
      [column_constraint],
      ...
      [table_constraint][,...]);
```

– Column-level constraint syntax:

```
column [CONSTRAINT constraint_name] constraint_type,
```

– Table-level constraint syntax:

```
column,...
  [CONSTRAINT constraint_name] constraint_type
  (column, ...),
```

Defining Constraints

– Example of a column-level constraint:

```
CREATE TABLE orders(
  order_id  NUMBER(4)
    CONSTRAINT ord_ord_id_pk PRIMARY KEY,
  order_mode    VARCHAR2(20),
  ...);
```

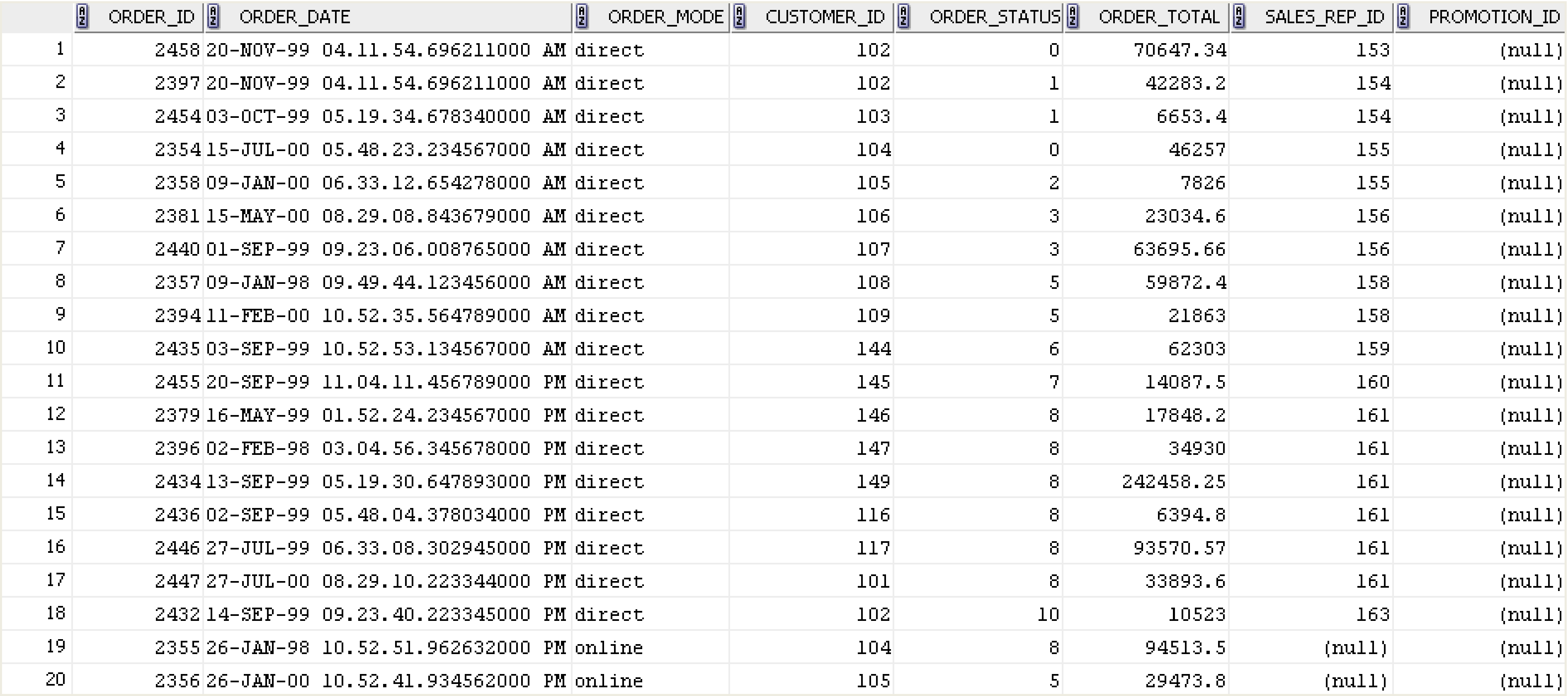
1

– Example of a table-level constraint:

```
CREATE TABLE orders(
  order_id  NUMBER(6),
  order_mode    VARCHAR2(20),
  ...
  customer_id  VARCHAR2(10) NOT NULL,
  CONSTRAINT ord_ord_id_pk
    PRIMARY KEY (ORDER_ID));
```

2

NOT NULL Constraint **Orders table**



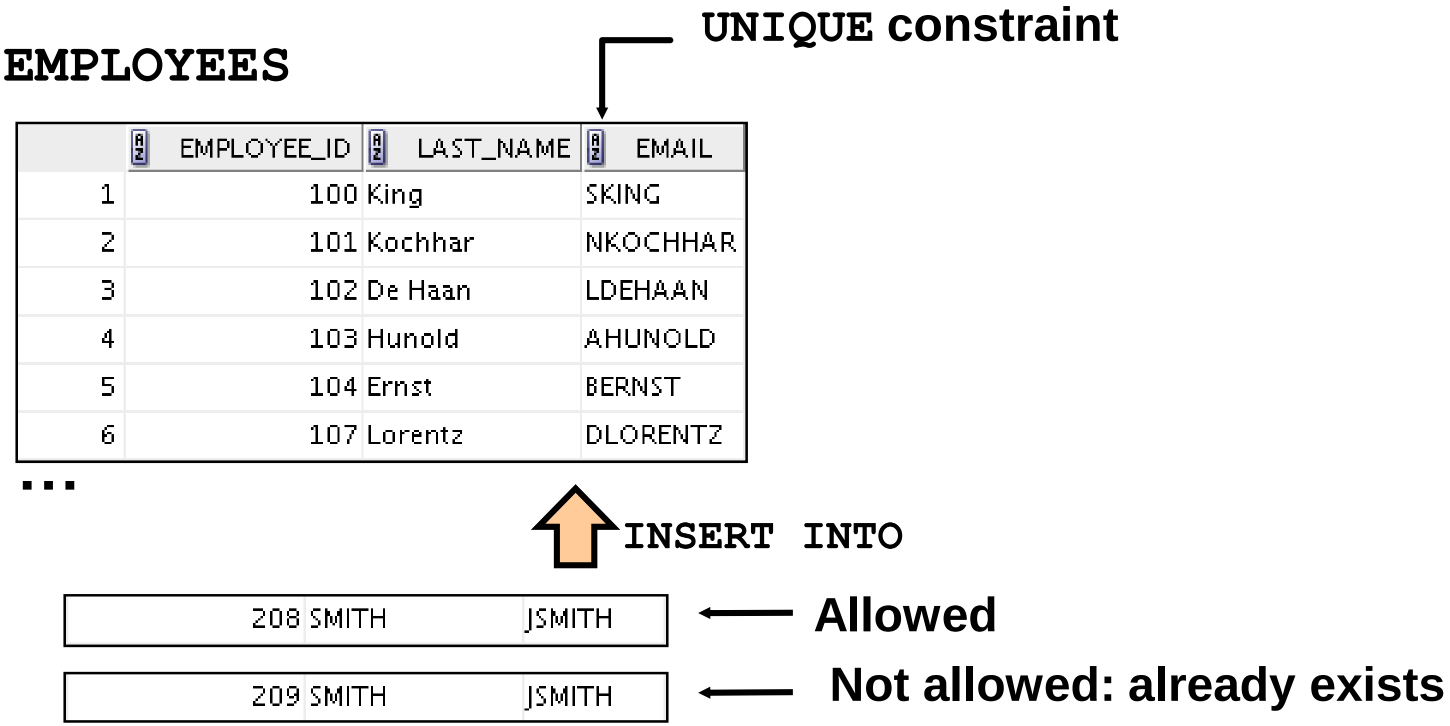
	ORDER_ID	ORDER_DATE	ORDER_MODE	CUSTOMER_ID	ORDER_STATUS	ORDER_TOTAL	SALES_REP_ID	PROMOTION_ID
1	2458	20-NOV-99 04.11.54.696211000 AM	direct	102	0	70647.34	153	(null)
2	2397	20-NOV-99 04.11.54.696211000 AM	direct	102	1	42283.2	154	(null)
3	2454	03-OCT-99 05.19.34.678340000 AM	direct	103	1	6653.4	154	(null)
4	2354	15-JUL-00 05.48.23.234567000 AM	direct	104	0	46257	155	(null)
5	2358	09-JAN-00 06.33.12.654278000 AM	direct	105	2	7826	155	(null)
6	2381	15-MAY-00 08.29.08.843679000 AM	direct	106	3	23034.6	156	(null)
7	2440	01-SEP-99 09.23.06.008765000 AM	direct	107	3	63695.66	156	(null)
8	2357	09-JAN-98 09.49.44.123456000 AM	direct	108	5	59872.4	158	(null)
9	2394	11-FEB-00 10.52.35.564789000 AM	direct	109	5	21863	158	(null)
10	2435	03-SEP-99 10.52.53.134567000 AM	direct	144	6	62303	159	(null)
11	2455	20-SEP-99 11.04.11.456789000 PM	direct	145	7	14087.5	160	(null)
12	2379	16-MAY-99 01.52.24.234567000 PM	direct	146	8	17848.2	161	(null)
13	2396	02-FEB-98 03.04.56.345678000 PM	direct	147	8	34930	161	(null)
14	2434	13-SEP-99 05.19.30.647893000 PM	direct	149	8	242458.25	161	(null)
15	2436	02-SEP-99 05.48.04.378034000 PM	direct	116	8	6394.8	161	(null)
16	2446	27-JUL-99 06.33.08.302945000 PM	direct	117	8	93570.57	161	(null)
17	2447	27-JUL-00 08.29.10.223344000 PM	direct	101	8	33893.6	161	(null)
18	2432	14-SEP-99 09.23.40.223345000 PM	direct	102	10	10523	163	(null)
19	2355	26-JAN-98 10.52.51.962632000 PM	online	104	8	94513.5	(null)	(null)
20	2356	26-JAN-00 10.52.41.934562000 PM	online	105	5	29473.8	(null)	(null)

NOT NULL constraint (Primary Key enforces NOT NULL constraint.)

NOT NULL constraint

Absence of NOT NULL constraint (Any row can contain a null value for this column.)

UNIQUE Constraint



EMPLOYEES

	EMPLOYEE_ID	LAST_NAME	EMAIL
1	100	King	SKING
2	101	Kochhar	NKOCHHAR
3	102	De Haan	LDEHAAN
4	103	Hunold	AHUNOLD
5	104	Ernst	BERNST
6	107	Lorentz	DLORENTZ

...

UNIQUE constraint

INSERT INTO

208	SMITH	JSMITH
209	SMITH	JSMITH

Allowed

Not allowed: already exists

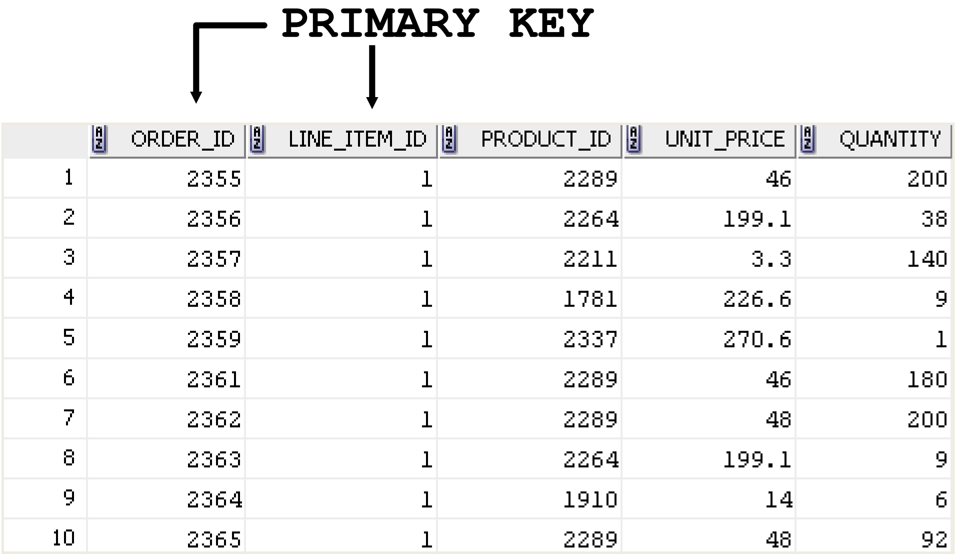
UNIQUE Constraint

Defined at either the table level or the column level:

```
CREATE TABLE orders(
    order_id            NUMBER(4),
    order_mode          VARCHAR2(25) NOT NULL,
    order_status        CHAR(2),
    customer_id         NUMBER(8,2),
    order_date          DATE NOT NULL,
...
    CONSTRAINT ord_id_uk UNIQUE(order_id));
```

PRIMARY KEY Constraint

ORDER_ITEMS



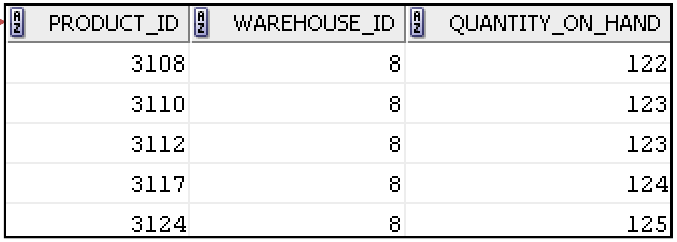
PRIMARY KEY

	ORDER_ID	LINE_ITEM_ID	PRODUCT_ID	UNIT_PRICE	QUANTITY
1	2355	1	2289	46	200
2	2356	1	2264	199.1	38
3	2357	1	2211	3.3	140
4	2358	1	1781	226.6	9
5	2359	1	2337	270.6	1
6	2361	1	2289	46	180
7	2362	1	2289	48	200
8	2363	1	2264	199.1	9
9	2364	1	1910	14	6
10	2365	1	2289	48	92

FOREIGN KEY Constraint

INVENTORIES TABLE

PRIMARY KEY

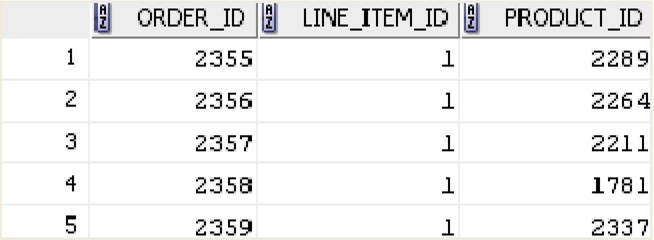


PRODUCT_ID	WAREHOUSE_ID	QUANTITY_ON_HAND
3108	8	122
3110	8	123
3112	8	123
3117	8	124
3124	8	125

...

ORDER_ITEMS TABLE

FOREIGN KEY



	ORDER_ID	LINE_ITEM_ID	PRODUCT_ID
1	2355	1	2289
2	2356	1	2264
3	2357	1	2211
4	2358	1	1781
5	2359	1	2337

...

FOREIGN KEY Constraint

Defined at either the table level or the column level:

```
CREATE TABLE orders(
    order_id         NUMBER(4),
    order_mode       VARCHAR2(25) NOT NULL,
    order_status     CHAR(2),
    customer_id      NUMBER(8,2),
    order_date       DATE NOT NULL,
...

CONSTRAINT ord_inv_fk FOREIGN KEY (order_id)
      REFERENCES inventories(order_id));
```

FOREIGN KEY Constraint: Keywords

FOREIGN KEY: Defines the column in the child table at the table-constraint level

REFERENCES: Identifies the table and column in the parent table

ON DELETE CASCADE:

Deletes the dependent rows in the child table when a row in the parent table is deleted

ON DELETE SET NULL:

Converts dependent foreign key values to null

CHECK Constraint

Defines a condition that each row must satisfy

The following expressions are not allowed:

- References to CURRVAL, NEXTVAL, LEVEL, and ROWNUM pseudo columns
- Calls to SYSDATE, UID, USER, and USERENV functions
- Queries that refer to other values in other rows

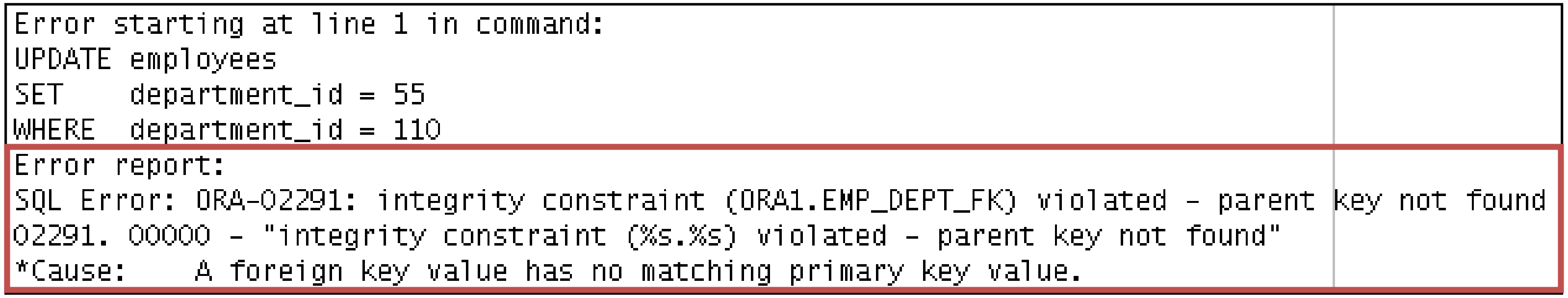
```
..., order_status NUMBER(2)
     CONSTRAINT ord_status_btw
          CHECK (order_status BETWEEN 0 AND 10),...
```

CREATE TABLE: Example

```
CREATE   TABLE   customers

            ( customer_id           NUMBER(6)
            , cust_first_name       VARCHAR2(20)
                 CONSTRAINT cust_fname_nn NOT NULL
            , cust_last_name        VARCHAR2(20)
                 CONSTRAINT cust_lname_nn NOT NULL
            , cust_address          cust_address_typ
            , phone_numbers         phone_list_typ
            , nls_language          VARCHAR2(3)
            , nls_territory         VARCHAR2(30)
            , credit_limit          NUMBER(9,2)
            , cust_email            VARCHAR2(30)
            , account_mgr_id        NUMBER(6)
            ,      CONSTRAINT          customer_credit_limit_max
                               CHECK (credit_limit <= 5000)
            ,      CONSTRAINT          customer_id_min
                               CHECK (customer_id > 0)
    ) ;
```

Violating Constraints

```
UPDATE employees
SET    department_id = 55
WHERE  department_id = 110;
```



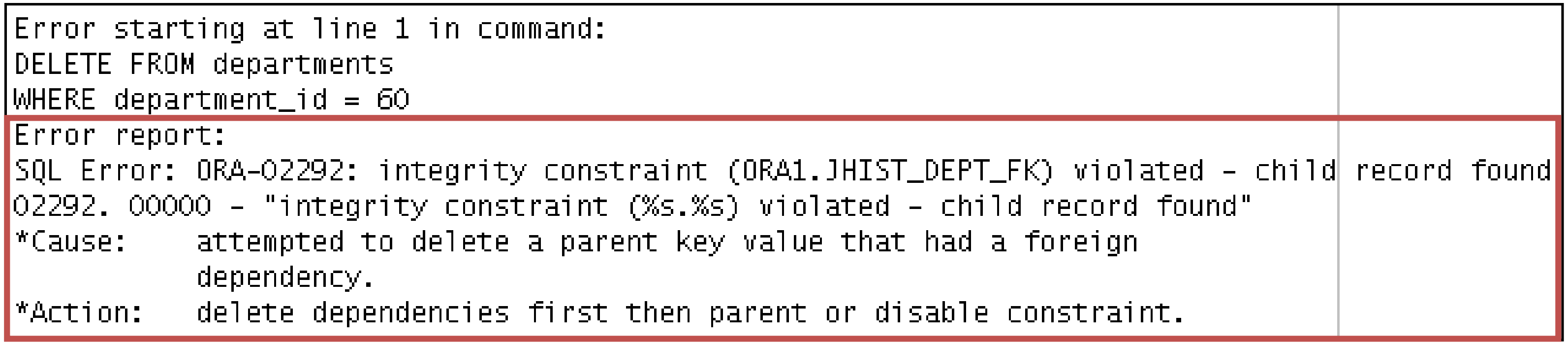
```
Error starting at line 1 in command:
UPDATE employees
SET    department_id = 55
WHERE  department_id = 110
Error report:
SQL Error: ORA-02291: integrity constraint (ORA1.EMP_DEPT_FK) violated - parent key not found
02291. 00000 - "integrity constraint (%s.%s) violated - parent key not found"
*Cause:    A foreign key value has no matching primary key value.
```

Department 55 does not exist.

Violating Constraints

You cannot delete a row that contains a primary key that is used as a foreign key in another table.

```
DELETE FROM departments
WHERE department_id = 60;
```



```
Error starting at line 1 in command:
DELETE FROM departments
WHERE department_id = 60
Error report:
SQL Error: ORA-02292: integrity constraint (ORA1.JHIST_DEPT_FK) violated - child record found
02292. 00000 - "integrity constraint (%s.%s) violated - child record found"
*Cause:    attempted to delete a parent key value that had a foreign
           dependency.
*Action:   delete dependencies first then parent or disable constraint.
```

Creating a Table Using a Subquery

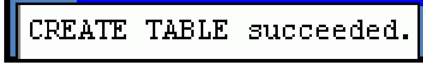
- Create a table and insert rows by combining the CREATE TABLE statement and the AS *subquery* option.

```
CREATE TABLE table
           [(column, column...)]
AS subquery;
```

- Match the number of specified columns to the number of subquery columns.
- Define columns with column names and default values.

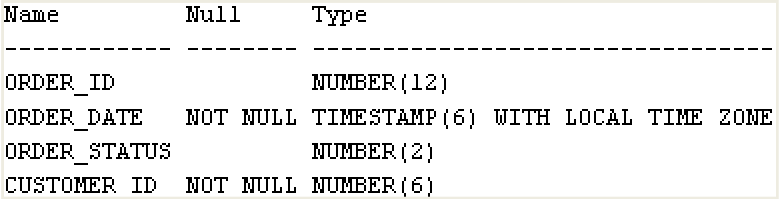
Creating a Table Using a Subquery

```
CREATE  TABLE   ord2458
  AS
    SELECT  order_id ,  order_date ,
            order_status ,
            customer_id
    FROM   orders
    WHERE   order_id  =  2458 ;
```



```
CREATE TABLE succeeded.
```

```
DESCRIBE   ord2458 ;
```



```
Name         Null     Type
------------ -------- ---------------------------------
ORDER_ID              NUMBER(12)
ORDER_DATE   NOT NULL TIMESTAMP(6) WITH LOCAL TIME ZONE
ORDER_STATUS          NUMBER(2)
CUSTOMER_ID  NOT NULL NUMBER(6)
```

Dropping a Table

Moves a table to the recycle bin

Removes the table and all its data entirely if the PURGE clause is specified

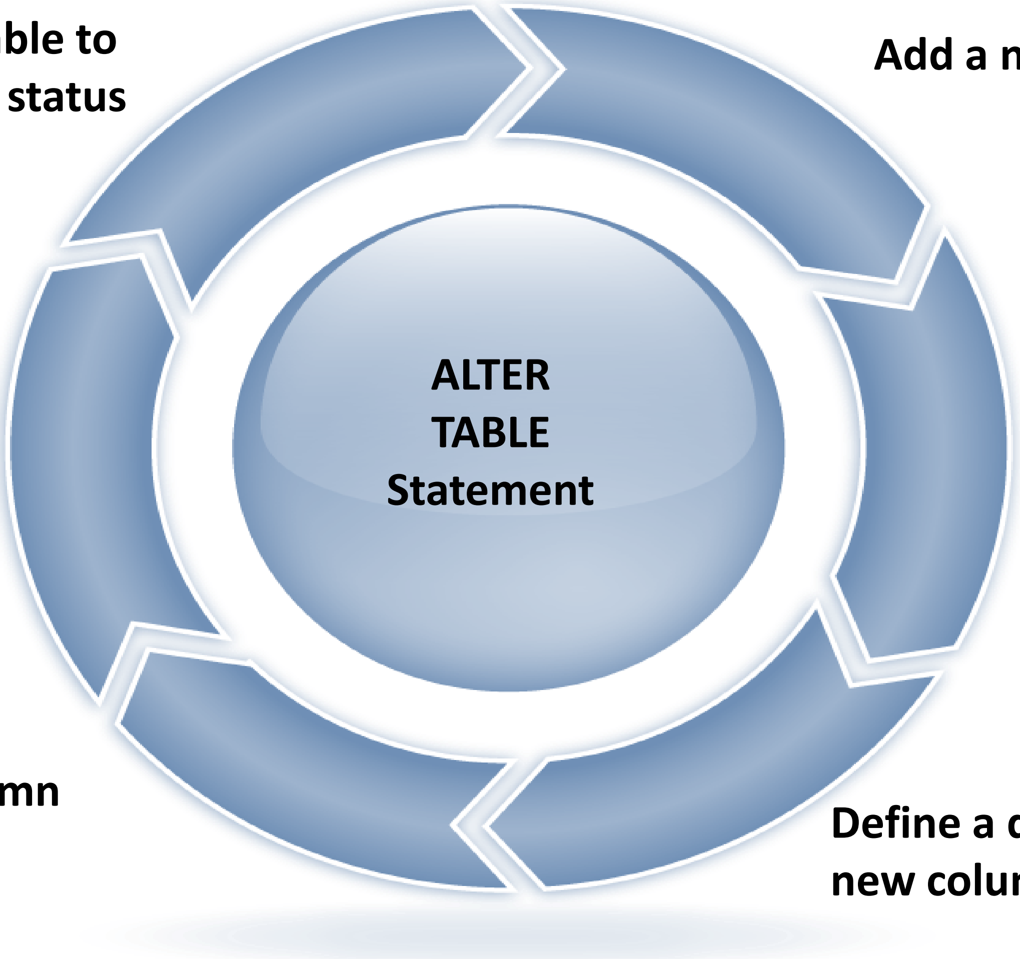
Invalidates dependent objects and removes object privileges on the table

```
DROP TABLE ord2458;
```

```
DROP TABLE dept80 succeeded.
```

ALTER TABLE Statement

Use the ALTER TABLE statement to:



Change table to read-only status

Add a new column

Rename a column

ALTER TABLE Statement

Modify an existing column definition

Drop a column

Define a default value for the new column

Read-Only Tables

You can use the ALTER TABLE syntax to:

- Put a table into read-only mode, which prevents DDL or DML changes during table maintenance
- Put the table back into read/write mode

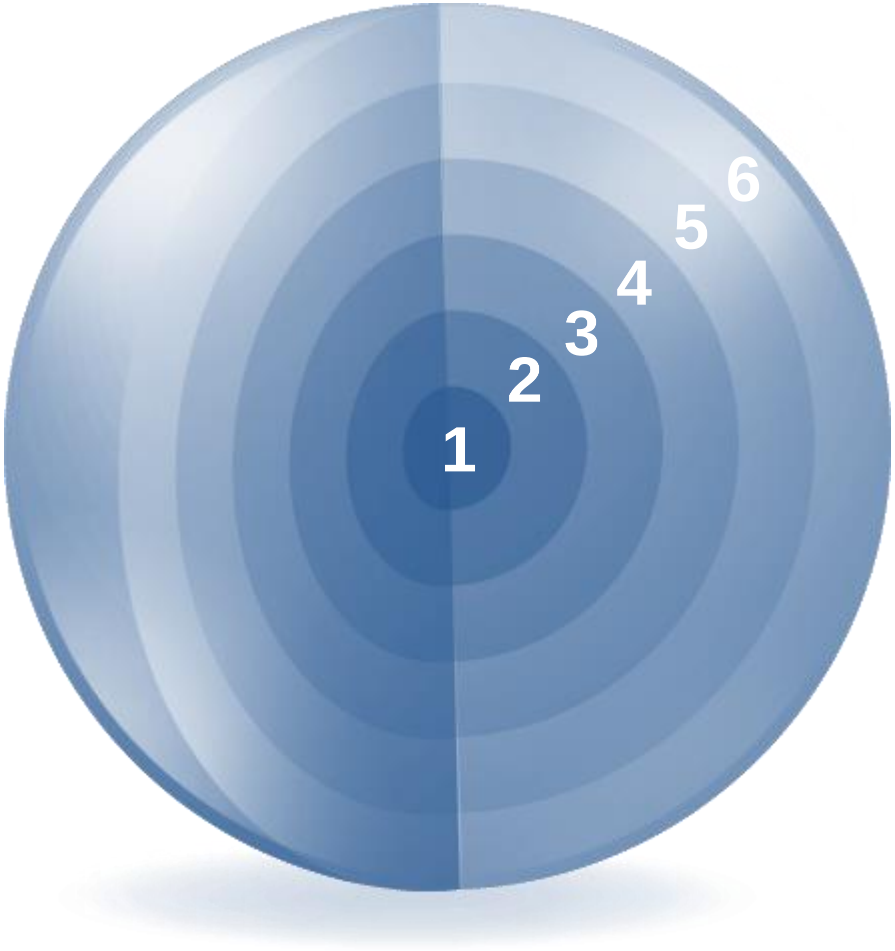
```
ALTER   TABLE  orders  READ  ONLY ;

-- perform table maintenance and then
-- return table back to read/write mode

ALTER   TABLE  orders  READ  WRITE ;
```

Quiz

You can use constraints to do the following:

1. Enforce rules on the data in a table whenever a row is inserted, updated, or deleted.
2. Prevent the deletion of a table.
3. Prevent the creation of a table.
4. Prevent the creation of data in a table.

Session Summary



1. Categorize the main database objects

2. Review the table structure

3. List the data types that are available for columns

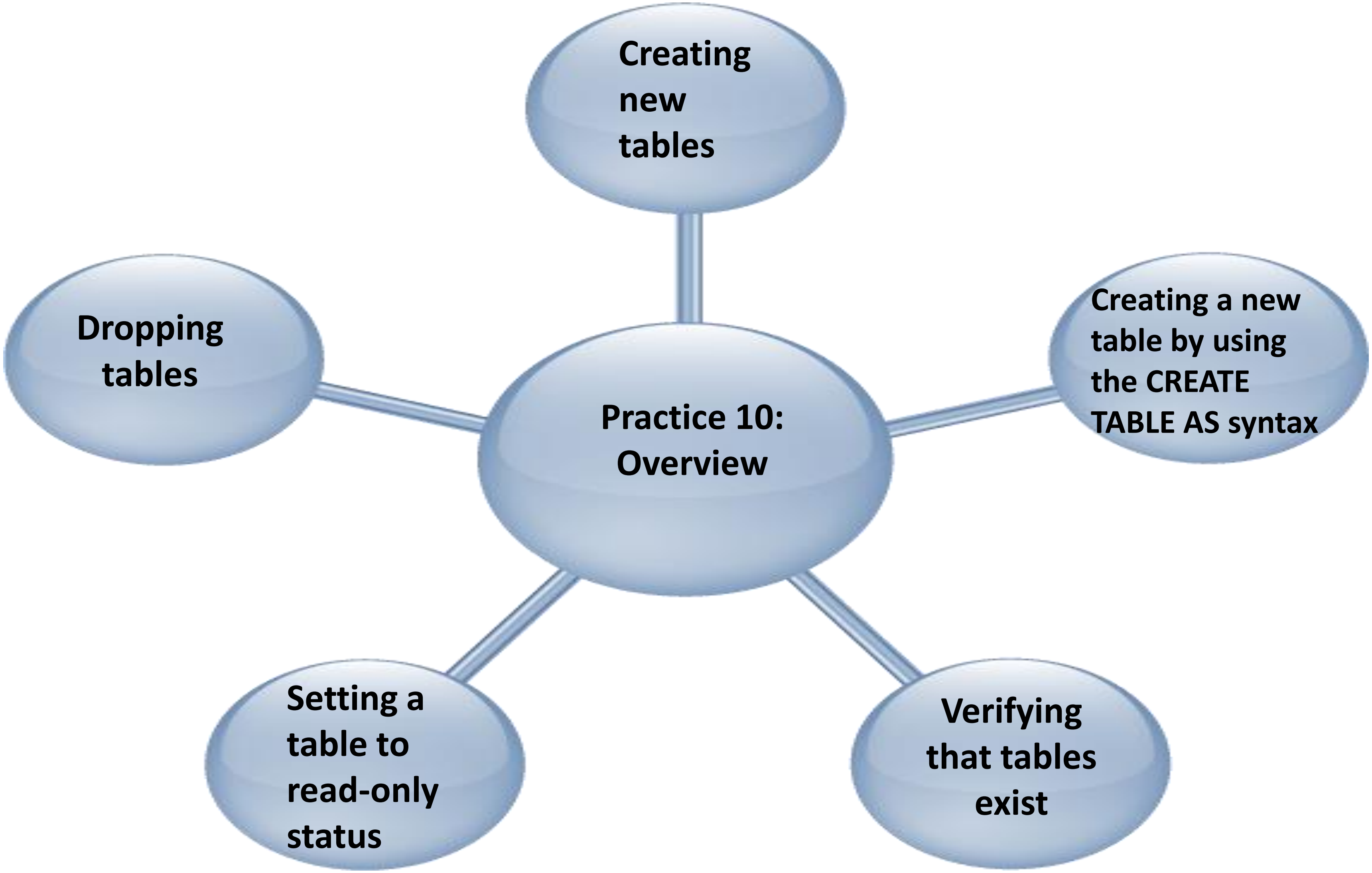
.

4. Create a simple table

5. Explain how constraints are created at the time of table creation

6. Describe how schema objects work

Practice 10: Overview



Practice 10: Overview

Creating new tables

Creating a new table by using the CREATE TABLE AS syntax

Verifying that tables exist

Setting a table to read-only status

Dropping tables