RELATIONAL DATABASES

OBJECTIVES

- DATABASE OBJECTS
- TABLE STRUCTURE
- CREATE TABLE
- DATA DICTIONARY

PURPOSE

- YOUR TABLES WILL BE SMALL COMPARED TO TABLES THAT HOLD MILLIONS OF ROWS AND HUNDREDS OF COLUMNS, BUT CREATING A SMALL TABLE INVOLVES THE SAME SQL STATEMENTS AND SYNTAX AS CREATING A VERY LARGE TABLE.

SCHEMA OBJECTS

- TABLE
- INDEX
- CONSTRAINT
- VIEW
- SEQUENCE
- SYNONYM

SCHEMA OBJECTS

- SOME OBJECT TYPES TAKE UP SPACE, KNOWN AS STORAGE, OTHERS DO NOT.
- DATABASE OBJECTS TAKING UP SIGNIFICANT STORAGE SPACE ARE KNOWN AS SEGMENTS.
- TABLES AND INDEXES ARE EXAMPLES OF SEGMENTS AS THE VALUES STORED IN THE COLUMNS OF EACH ROW TAKE UP SIGNIFICANT PHYSICAL DISK SPACE.
- VIEWS, CONSTRAINTS, SEQUENCES AND SYNONYMS ARE ALSO OBJECTS BUT THE ONLY SPACE THEY REQUIRE IN THE DATABASE IS IN THE DEFINITION OF THE OBJECT – NONE OF THEM HAVE ANY DATA ROWS ASSOCIATED WITH THEM.

SCHEMA OBJECTS

- THE DATABASE STORES THE DEFINITIONS OF ALL DATABASE OBJECTS IN THE DATA DICTIONARY, AND THESE DEFINITIONS ARE ACCESSIBLE TO ALL USERS OF THE DATABASE AS WELL AS TO THE DATABASE ITSELF.
- THE DATABASE LOOKS UP THE DATA DICTIONARY AND FINDS THE DEFINITION OF THE TABLE USED IN THE QUERY.
- THE DATA DICTIONARY IS USED FOR ALL STATEMENTS ISSUED.
- IT CHECKS THAT THE TABLES YOU ARE REFERENCING IN YOUR STATEMENT EXIST IN THE DATABASE, IT CHECKS THE COLUMN NAMES ARE CORRECT, AND THAT YOU HAVE THE CORRECT PRIVILEGES TO PERFORM THE ACTION YOU ARE REQUESTING AND FINALLY IT USES THE DICTIONARY TO DECIDE THE EXECUTION PLAN – HOW IT WILL PERFORM THE REQUEST.

SCHEMA OBJECTS

- THE DATA DICTIONARY CAN BE QUERIED BY ALL DATABASE USERS.
- IN APPLICATION EXPRESS, IT CAN BE ACCESSED BOTH VIA SQL STATEMENTS IN THE SQL WORKSHOP>SQL COMMAND INTERFACE AND ALSO FROM THE SQL WORKSHOP>OBJECT BROWSER INTERFACE.
- IN THE SQL COMMANDS WINDOW, YOU HAVE TO KNOW THE NAMES OF THE TABLE YOU ARE QUERYING. YOU USE THE DESC COMMAND TO VIEW A TABLES DECRIPTION.
- IN THE OBJECT BROWSER INTERFACE YOU SIMPLY CLICK THE LISTED OBJECTS TO SEE THEIR DETAILS.

DATA DICTIONARY

- TWO KINDS OF TABLES EXIST IN THE DATA DICTIONARY, USER TABLES AND DATA DICTIONARY TABLES.
- YOU CAN USE SQL STATEMENTS TO ACCESS BOTH KINDS OF TABLES – YOU CAN SELECT, INSERT, UPDATE, AND DELETE DATA IN THE USER TABLES, AND YOU CAN SELECT DATA IN THE DATA DICTIONARY TABLES.
- USER TABLES: EMPLOYEES, DEPARTMENTS, JOBS ETC
- DATA DICTIONARY TABLES: DICTIONARY, USER_OBJECTS, USER_TABLES, USER_SEGMENTS, USER_INDEXES ETC
- DATA DICTIONARY TABLES ARE ALL OWNED BY A SPECIAL ORACLE USER CALLED SYS AND ONLY SELECT STATEMENTS SHOULD BE USED WITH ANY OF THESE TABLES.

DATA DICTIONARY

- WHEN YOU ARE USING THE DATA DICTIONARY YOU MUST KNOW THE NAMES OF WHAT YOU WISH TO SEE.
- IN ORACLE THIS IS QUITE SIMPLE: PREFIX THE OBJECT TYPE YOU ARE LOOKING FOR WITH A USER_XXX OR AND ALL_XXX, WHERE XXX IS THE OBJECT TYPE.

```
SELECT table_name, status
FROM USER_TABLES;
```

```
SELECT table_name, status
FROM ALL_TABLES;
```

DATA DICTIONARY

- SO IF YOU WANT TO INVESTIGATE INDEXES, THEN SIMPLY SELECT FROM USER_INDEXES, IF YOU WANT INFORMATION ABOUT SEQUENCES, THEN THE TABLE IS USER_SEQENCES AND SO ON

```
SELECT *
FROM user_indexes;
```

```
SELECT *
FROM user_objects
WHERE object_type = 'SEQUENCE';
```

TABLE CREATION

- ALL DATA IN A RELATIONAL DATABASE IS STORED IN TABLES.
- WHEN CREATING A NEW TABLE, USE THE FOLLOWING RULES FOR NAMES:
  - MUST BEGIN WITH A LETTER.
  - MUST BE 1 TO 30 CHARACTERS LONG
  - MUST CONTAIN ONLY A-Z, A-Z, 0-9, _, $, AND #
  - MUST NOT DUPLICATE THE NAME OF ANOTHER OBJECT OWNED BY THE SAME USER
  - MUST NOT BE AN ORACLE SERVER RESERVED WORD.
- TABLE NAMES ARE NOT CASE SENSITIVE, THEY SHOULD BE PLURAL

DATA DEFINITION LANGUAGE (DDL)

- CREATING TABLES IS PART OF SQL'S DATA DEFINITION LANGUAGE (DDL).
- OTHER DDL STATEMENTS USED TO SET UP, CHANGE, AND REMOVE DATA STRUCTURES FROM TABLES INCLUDE ALTER, DROP, RENAME, AND TRUNCATE.
- TO CREATE A NEW TABLE, YOU MUST HAVE THE CREATE TABLE PRIVILEGE AND A STORAGE AREA FOR IT.
- THE DATABASE ADMINISTRATOR USES DATA CONTROL LANGUAGE (DCL) STATEMENTS TO GRANT THIS PRIVILEGE TO USERS AND ASSIGN A STORAGE AREA.
- TABLES BELONGING TO OTHER USERS ARE NOT IN ;YOUR SCHEMA (STORAGE AREA).
- IF YOU WANT TO VIEW A TABLE THAT IS NOT IN YOUR SCHEMA YOU USE THE TABLE OWNERS NAME AS A PREFIX TO THE TABLE NAME.

CREAT TABLE SYNTAX

- TO CREATE A NEW TABLE, USE THE FOLLOWING SYNTAX DETAILS:
  - TABLE IS THE NAME OF THE TABLE.
  - COLUMN IS THE NAME OF THE COLUMN
  - DATA TYPE IS THE COLUMN'S DATA TYPE AND LENGTH
  - DEFAULT EXPRESSION SPECIFIES A DEFAULT VALUE IF A VALUE IS OMITTED WHEN DATA IS INSERTED.

```
CREATE TABLE table
(column data type [DEFAULT expression],
(column data type [DEFAULT expression],
(……[ ] );
```

CREATE TABLE EXAMPLE

```
CREATE TABLE my_cd_collection
 (cd_number NUMBER(3),
  title VARCHAR2(20),
  artist VARCHAR2(20),
  purchase_date DATE DEFAULT SYSDATE);
```

```
CREATE TABLE my_friends
 (first_name VARCHAR2(20),
  last_name VARCHAR2(30),
  email VARCHAR2(30),
  phone_num VARCHAR2(12),
  birth_date DATE);
```