CSE 462 – Introduction to Oracle

CREATE TABLE: Basic Syntax

```
CREATE TABLE <table_name> (
    <column_name> <column_type> [<inline_constraints>],
    ...
    <column_name> <column_type> [<inline_constraints>] [,
    <table_constraint1> [,
    ...
    <table_constraintN>] ...]
);
```

CSE 462 – Introduction to Oracle

Table and Column Names

- Must start with a letter
- Must contain only letters, digits, and underscores
- Must be up to 30 characters in length

CSE 462 – Introduction to Oracle

Basic Character Data Types

- CHAR(size): fixed length (<= 2000) character strings
- VARCHAR2(size): variable length (<= 4000) character strings
- CLOB: large character objects (128 terabytes)
- there are corresponding versions for Unicode strings:
  - NCHAR
  - NVARCHAR2
  - NCLOB

CSE 462 – Introduction to Oracle

Basic Numeric Data Types

- NUMBER(precision, scale): up to 38 significant digits
- NUMBER: if you do not want to specify precision
- INTEGER: integer with maximum precision of 38 significant digits

CSE 462 – Introduction to Oracle

Other Data Types

- DATE: date values, but no time component
- TIMESTAMP: date and time values
- BLOB: binary large object
- BFILE: a file outside the database

CSE 462 – Introduction to Oracle

Example

```
CREATE TABLE STATE (
    State          CHAR(2),
    State_Name     VARCHAR(30));
```

CSE 462 – Introduction to Oracle

Example

```
CREATE TABLE CITY (
    City_ID       NUMBER,
    City_Name     VARCHAR(100),
    State         CHAR(2),
    Foundation    DATE,
    Population    NUMBER);
```

CSE 462 – Introduction to Oracle

General Form of a Table Constraint

```
CONSTRAINT <constraint_name> <constraint_type> <constraint_specs>
```

Some Constraint Types

- PRIMARY KEY
- UNIQUE
- CHECK
- FOREIGN KEY

CSE 462 – Introduction to Oracle

PRIMARY KEY

- At most one per table
- Primary key fields are not NULLable
- Duplicate combinations of key field values not allowed

CSE 462 – Introduction to Oracle

UNIQUE CONSTRAINT

- Unique key fields may be NULLable
- Duplicate combinations of key field values not allowed
- NULL values are not considered equal in this comparison
- Oracle non-standard behavior regarding NULL values:
  - Consider a table with two NULLable INT fields A, B
    - Record (NULL, 1) allowed at most ONCE
    - Record (1, NULL) allowed at most ONCE
    - Record (NULL, NULL) allowed ANY NUMBER OF TIMES

CSE 462 – Introduction to Oracle

FOREIGN KEY

- Foreign key fields may be NULLable
- The number and type of the constrained fields must match the the number and type of the referenced fields
- Foreign key field values must match the values of the referenced fields in some row of the referenced table

CSE 462 – Introduction to Oracle

CHECK CONSTRAINT

- Checked fields may be NULLable
- A check expression is a boolean expression involving one or more fields
- A check constraint is satisfied if the check expression evaluates to true or the NULL value
- Since most expressions will evaluate to NULL if any operand is NULL, they will not prevent NULL values in the constrained columns

CSE 462 – Introduction to Oracle

Example: Primary Key Constraint

```
CREATE TABLE CITY (
    City_ID       NUMBER,
    City_Name     VARCHAR(100),
    State         CHAR(2),
    Foundation    DATE,
    Population    NUMBER,
    CONSTRAINT PK_CITY PRIMARY KEY (City_ID));
```

CSE 462 – Introduction to Oracle

Example: Unique Constraint

```
CREATE TABLE CITY (
    City_ID       NUMBER,
    City_Name     VARCHAR(100),
    State         CHAR(2),
    Foundation    DATE,
    Population    NUMBER,
    CONSTRAINT PK_CITY PRIMARY KEY (City_ID),
    CONSTRAINT UC_CITY_STATE UNIQUE (City_Name, State));
```

CSE 462 – Introduction to Oracle

Example: Check Constraint

```
CREATE TABLE CITY (
    City_ID       NUMBER,
    City_Name     VARCHAR(100),
    State         CHAR(2),
    Foundation    DATE,
    Population    NUMBER,
    CONSTRAINT PK_CITY PRIMARY KEY (City_ID),
    CONSTRAINT UC_CITY_STATE UNIQUE (City_Name, State),
    CONSTRAINT CK_CITY_POPULATION CHECK (Population > 1000));
```

CSE 462 – Introduction to Oracle

Example: Foreign Key Constraint

```
CREATE TABLE CITY (
    City_ID       NUMBER,
    City_Name     VARCHAR(100),
    State         CHAR(2),
    Foundation    DATE,
    Population    NUMBER,
    CONSTRAINT PK_CITY PRIMARY KEY (City_ID),
    CONSTRAINT UC_CITY_STATE UNIQUE (City_Name, State),
    CONSTRAINT CK_CITY_POPULATION CHECK (Population > 1000),
    CONSTRAINT FK_CITY_STATE FOREIGN KEY (State)
       REFERENCES STATE(State));
```

CSE 462 – Introduction to Oracle

Inline Constraints

- Bound to a single column
- Supported constraints:

```
NOT NULL
PRIMARY KEY
UNIQUE
CHECK(<condition>)
REFERENCES <table_name> [(<column_list>)]
```

CSE 462 – Introduction to Oracle

Example: Inline Constraints

```
CREATE TABLE CITY (
    City_ID       NUMBER NOT NULL PRIMARY KEY,
    City_Name     VARCHAR(100) NOT NULL,
    State         CHAR(2) NOT NULL REFERENCES STATE(State),
    Foundation    DATE,
    Population    NUMBER NOT NULL CHECK (Population > 1000),
    CONSTRAINT UC_CITY_STATE UNIQUE (City_Name, State));
```

CSE 462 – Introduction to Oracle

Running SQL*Plus

```
> setenv ORACLE_HOME /util/oracle/product/current/client_1
> setenv TWO_TASK aos.buffalo.edu
> sqlplus
```

CSE 462 – Introduction to Oracle

Using SQL*Plus

1. Select all columns and rows from MY_TABLE:

```
SQL> SELECT <field_list | *> FROM MY_TABLE;
```

2. List all user tables:

```
SQL> SELECT TABLE_NAME FROM USER_TABLES;
```

3. See the definition of a table:

```
SQL> DESCRIBE <table_name>;
```

CSE 462 – Introduction to Oracle

Using SQL*Plus (continued)

4. Drop MY_TABLE:

```
SQL> DROP TABLE MY_TABLE;
```

5. Execute the SQL script file myFile.sql (in the current folder):

```
SQL> start myFile
or
SQL> @myFile
```

6. Exit from SQL*PLUS:

```
SQL> quit
```