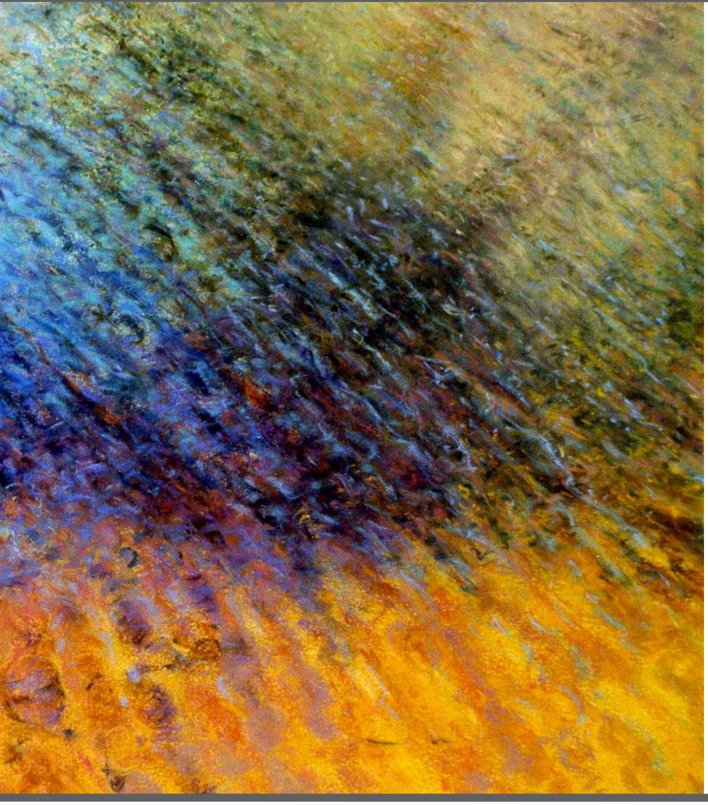
David M. Kroenke and David J. Auer

# Database Processing:

Fundamentals, Design, and Implementation



## Chapter Seven (Selected)

## SQL for Database Construction and Application Processing

# Database Available Online II

- For Microsoft SQL Server 2014:
  - See Online Chapter 10A
- For Oracle Database 12c and Oracle Database XE:
  - See Online Chapter 10B
- For MySQL 5.6
  - See Online Chapter 10C

- Online chapter 10B: Posted under “Oracle”

# SQL Categories

- SQL statements can be divided into five categories:
  - **Data definition language (DDL)**
  - **Data manipulation language (DML)** statements
  - ~~**SQL/Persistent Stored Modules (SQL/PSM)** statements~~
  - ~~**Transaction control language (TCL)** statements~~
  - ~~**Data control language (DCL)**~~ statements

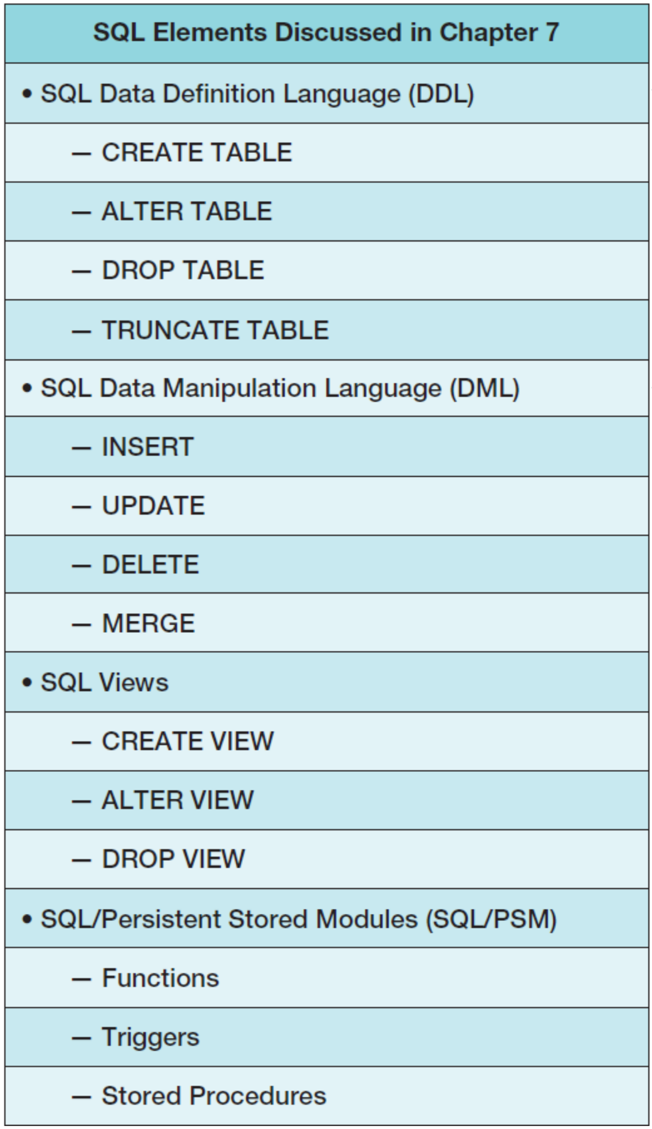
# SQL DDL

- **Data definition language (DDL)** statements
  - Used for creating tables
  - Covered in this chapter (Chapter 7)

# SQL DML

- **Data manipulation language (DML)** statements
  - Used for:
    - Queries – SQL **SELECT** statement
    - Inserting data – SQL **INSERT** statement
    - Modifying data – SQL **UPDATE** statement
    - Deleting data – SQL **DELETE** statement
  - Covered in Chapter 2

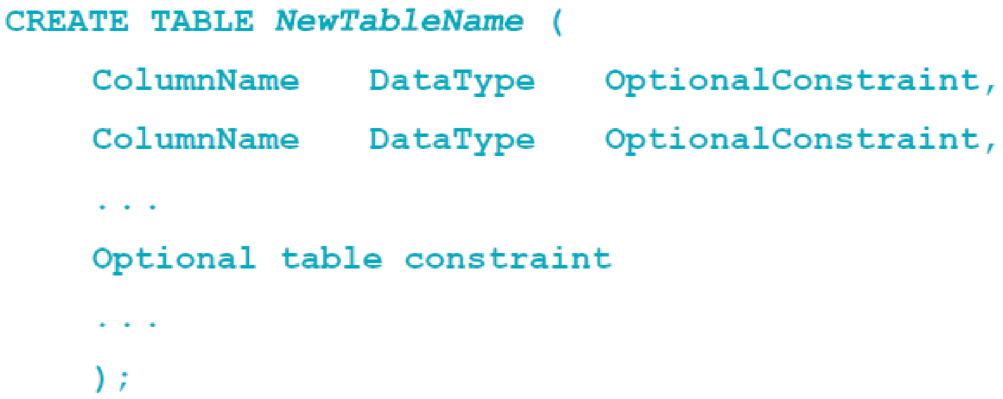
# Chapter 7 SQL Elements



| SQL Elements Discussed in Chapter 7 |
|---|
| • SQL Data Definition Language (DDL) |
| – CREATE TABLE |
| – ALTER TABLE |
| – DROP TABLE |
| – TRUNCATE TABLE |
| • SQL Data Manipulation Language (DML) |
| – INSERT |
| – UPDATE |
| – DELETE |
| – MERGE |
| • SQL Views |
| – CREATE VIEW |
| – ALTER VIEW |
| – DROP VIEW |
| • SQL/Persistent Stored Modules (SQL/PSM) |
| – Functions |
| – Triggers |
| – Stored Procedures |

# SQL CREATE TABLE Statement

- CREATE TABLE statement is used for creating relations.
- Each column is described with three parts: column name, data type, and optional constraints.
- Format:



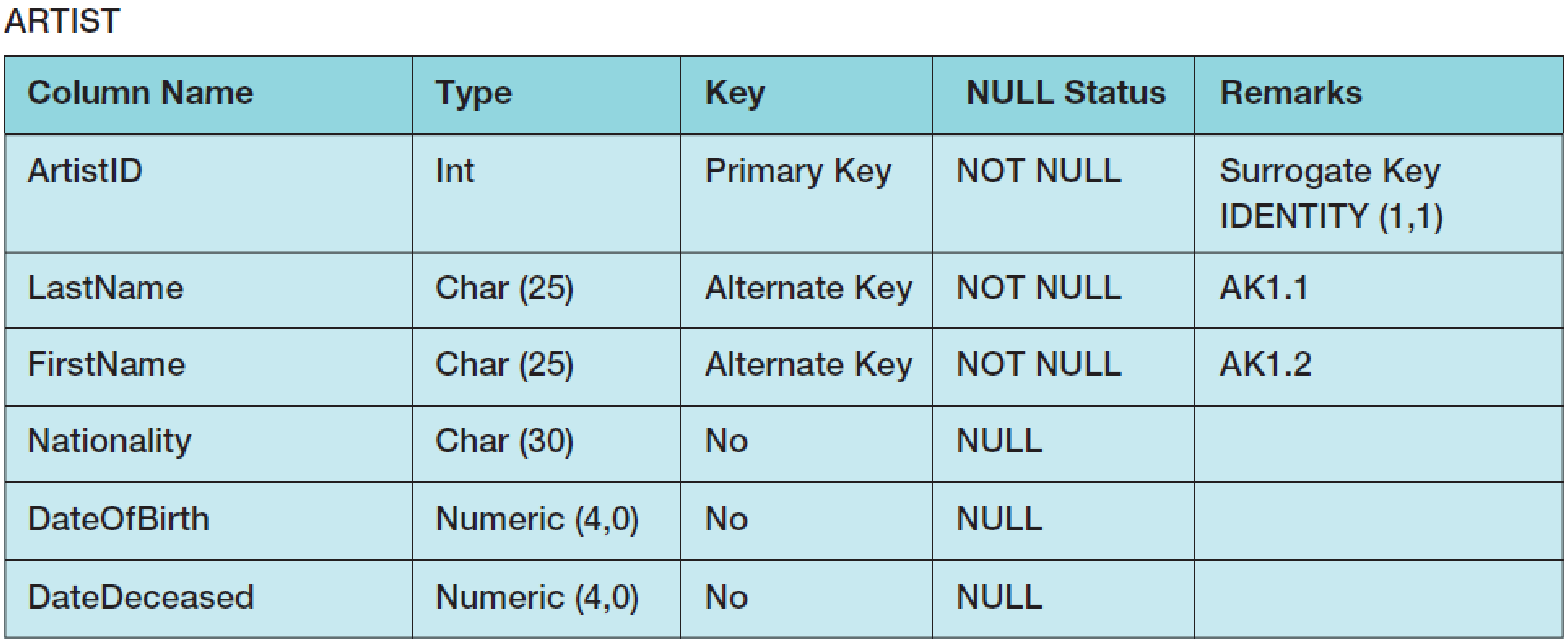
```
CREATE TABLE NewTableName (
    ColumnName    DataType    OptionalConstraint,
    ColumnName    DataType    OptionalConstraint,
    . . .
    Optional table constraint
    . . .
    );
```

# Column and Table Constraints

- Constraints can be defined within the CREATE TABLE statement (or added after using ALTER table statement).
- **Column and table constraints** include:
  - **PRIMARY KEY** – may not have NULL values
  - **FOREIGN KEY** – may not have NULL values
  - **NULL / NOT NULL**
  - **UNIQUE**
  - **CHECK**
- The **DEFAULT** keyword (not a constraint)

# SQL CREATE TABLE Statement Example I

## Column Characteristics:

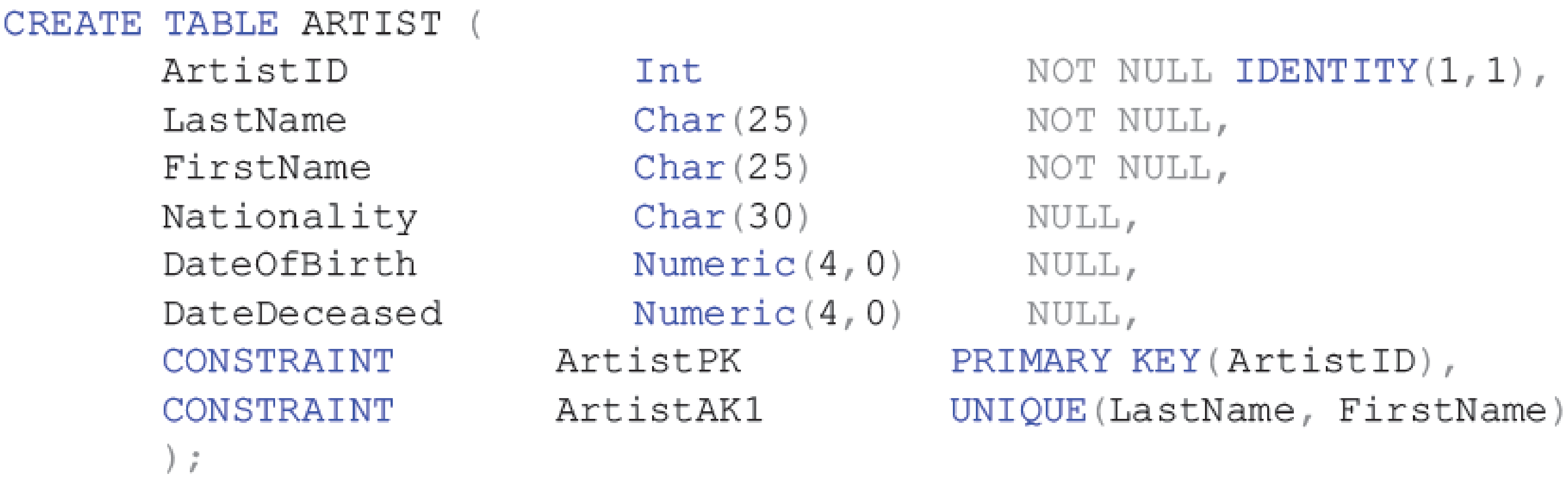


ARTIST

| Column Name | Type | Key | NULL Status | Remarks |
|---|---|---|---|---|
| ArtistID | Int | Primary Key | NOT NULL | Surrogate Key<br>IDENTITY (1,1) |
| LastName | Char (25) | Alternate Key | NOT NULL | AK1.1 |
| FirstName | Char (25) | Alternate Key | NOT NULL | AK1.2 |
| Nationality | Char (30) | No | NULL | |
| DateOfBirth | Numeric (4,0) | No | NULL | |
| DateDeceased | Numeric (4,0) | No | NULL | |

# SQL CREATE TABLE Statement Example II

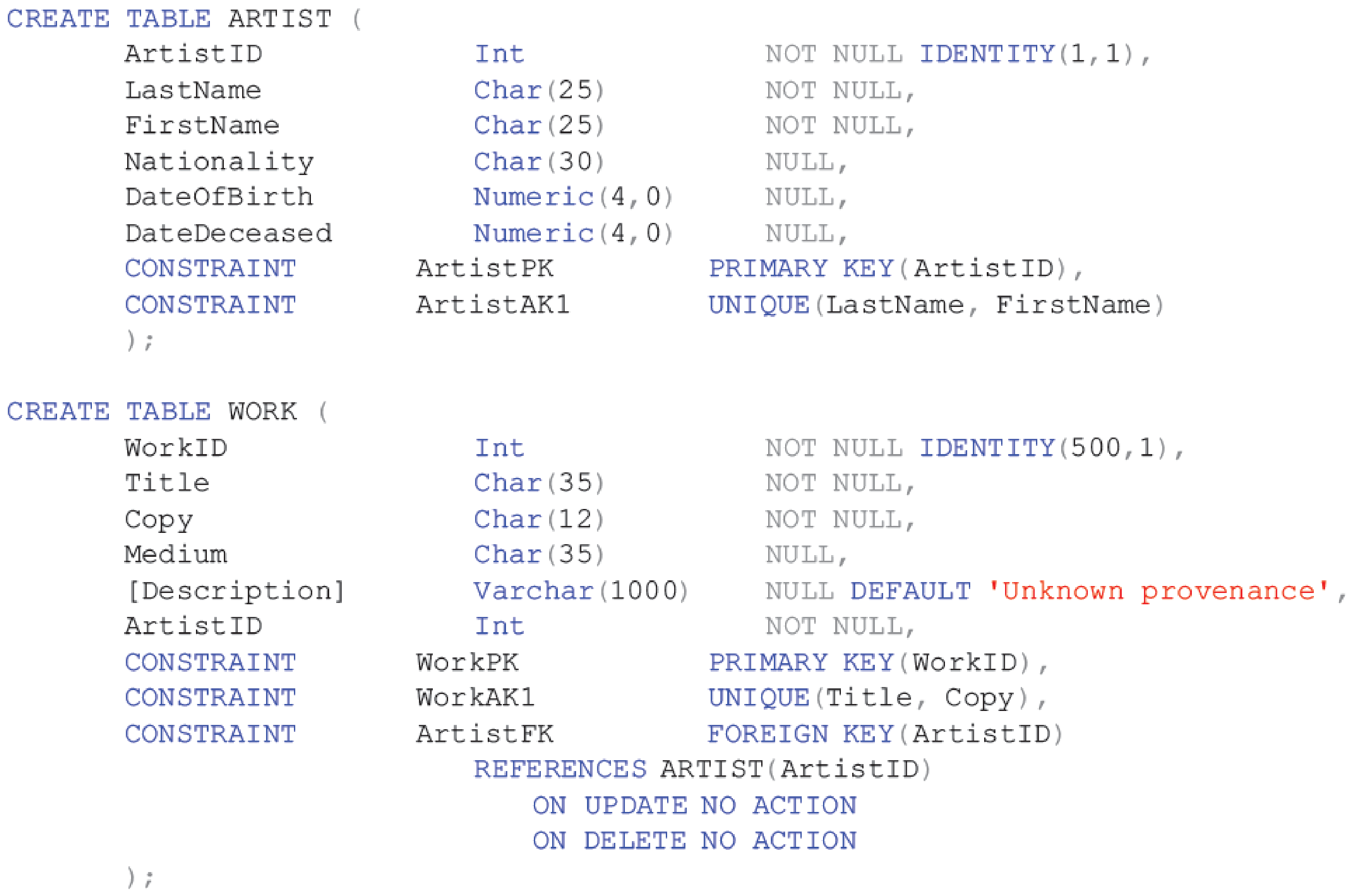
SQL CREATE TABLE statement:



```
CREATE TABLE ARTIST (
       ArtistID              Int                NOT NULL IDENTITY(1,1),
       LastName              Char(25)           NOT NULL,
       FirstName             Char(25)           NOT NULL,
       Nationality           Char(30)           NULL,
       DateOfBirth           Numeric(4,0)       NULL,
       DateDeceased          Numeric(4,0)       NULL,
       CONSTRAINT         ArtistPK           PRIMARY KEY(ArtistID),
       CONSTRAINT         ArtistAK1          UNIQUE(LastName, FirstName)
       );
```

Note: specific syntax for Oracle

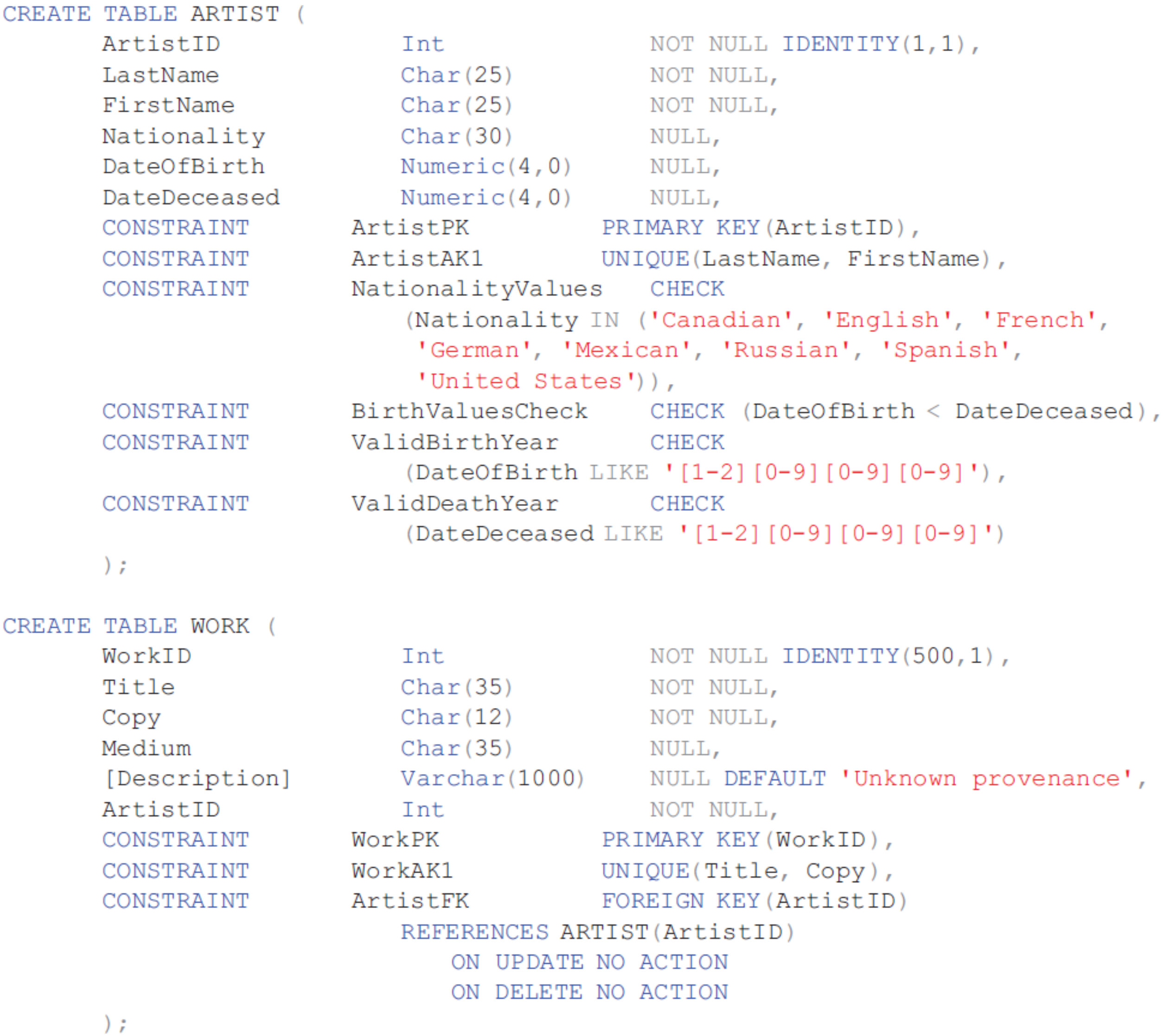
# Creating Relations



```
CREATE TABLE ARTIST (
      ArtistID            Int              NOT NULL IDENTITY(1,1),
      LastName            Char(25)         NOT NULL,
      FirstName           Char(25)         NOT NULL,
      Nationality         Char(30)         NULL,
      DateOfBirth         Numeric(4,0)     NULL,
      DateDeceased        Numeric(4,0)     NULL,
      CONSTRAINT       ArtistPK        PRIMARY KEY(ArtistID),
      CONSTRAINT       ArtistAK1       UNIQUE(LastName, FirstName)
      );

CREATE TABLE WORK (
      WorkID              Int              NOT NULL IDENTITY(500,1),
      Title               Char(35)         NOT NULL,
      Copy                Char(12)         NOT NULL,
      Medium              Char(35)         NULL,
      [Description]       Varchar(1000)    NULL DEFAULT 'Unknown provenance',
      ArtistID            Int              NOT NULL,
      CONSTRAINT       WorkPK          PRIMARY KEY(WorkID),
      CONSTRAINT       WorkAK1         UNIQUE(Title, Copy),
      CONSTRAINT       ArtistFK        FOREIGN KEY(ArtistID)
                          REFERENCES ARTIST(ArtistID)
                             ON UPDATE NO ACTION
                             ON DELETE NO ACTION
      );
```

# SQL for Constraints



```
CREATE TABLE ARTIST (
          ArtistID              Int                NOT NULL IDENTITY(1,1),
          LastName              Char(25)           NOT NULL,
          FirstName             Char(25)           NOT NULL,
          Nationality           Char(30)           NULL,
          DateOfBirth           Numeric(4,0)       NULL,
          DateDeceased          Numeric(4,0)       NULL,
          CONSTRAINT       ArtistPK          PRIMARY KEY(ArtistID),
          CONSTRAINT       ArtistAK1         UNIQUE(LastName, FirstName),
          CONSTRAINT       NationalityValues    CHECK
                               (Nationality IN ('Canadian', 'English', 'French',
                                'German', 'Mexican', 'Russian', 'Spanish',
                                'United States')),
          CONSTRAINT       BirthValuesCheck     CHECK (DateOfBirth < DateDeceased),
          CONSTRAINT       ValidBirthYear       CHECK
                               (DateOfBirth LIKE '[1-2][0-9][0-9][0-9]'),
          CONSTRAINT       ValidDeathYear       CHECK
                               (DateDeceased LIKE '[1-2][0-9][0-9][0-9]')
          );

CREATE TABLE WORK (
          WorkID                Int                NOT NULL IDENTITY(500,1),
          Title                 Char(35)           NOT NULL,
          Copy                  Char(12)           NOT NULL,
          Medium                Char(35)           NULL,
          [Description]         Varchar(1000)      NULL DEFAULT 'Unknown provenance',
          ArtistID              Int                NOT NULL,
          CONSTRAINT       WorkPK            PRIMARY KEY(WorkID),
          CONSTRAINT       WorkAK1           UNIQUE(Title, Copy),
          CONSTRAINT       ArtistFK          FOREIGN KEY(ArtistID)
                              REFERENCES ARTIST(ArtistID)
                                   ON UPDATE NO ACTION
                                   ON DELETE NO ACTION
          );
```

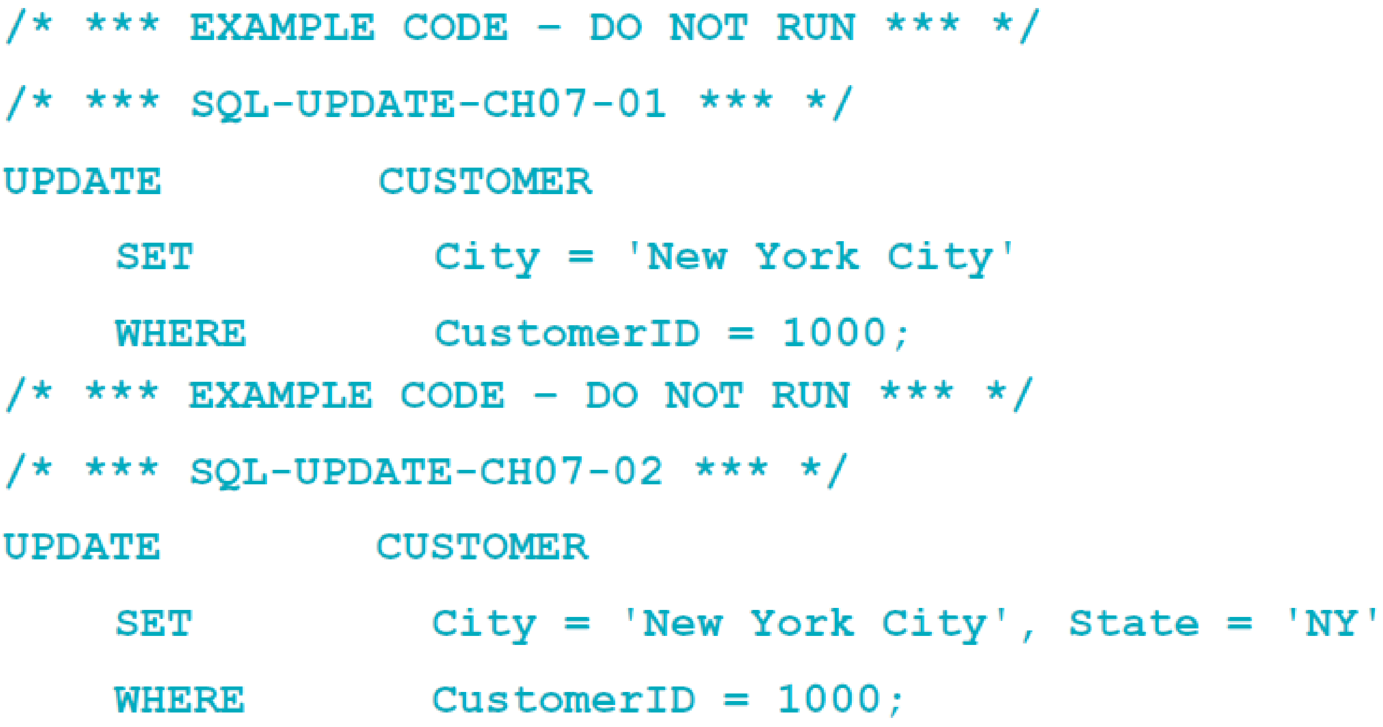
# SQL DML—INSERT I

- The **SQL INSERT statement**:

```
/* *** EXAMPLE CODE – DO NOT RUN *** */
/* *** SQL-INSERT-CH07-01 *** */
INSERT INTO ARTIST
     (LastName, FirstName, Nationality, DateOfBirth, DateDeceased)
     VALUES ('Miro', 'Joan', 'Spanish', 1893, 1983);
/* *** EXAMPLE CODE – DO NOT RUN *** */
/* *** SQL-INSERT-CH07-02 *** */
INSERT INTO ARTIST VALUES
     ('Miro', 'Joan', 'Spanish', 1893, 1983);
/* *** EXAMPLE CODE – DO NOT RUN *** */
/* *** SQL-INSERT-CH07-04 *** */
INSERT INTO ARTIST
     (LastName, FirstName, Nationality)
     VALUES ('Miro', 'Joan', 'Spanish');
```

# SQL DML—UPDATE I

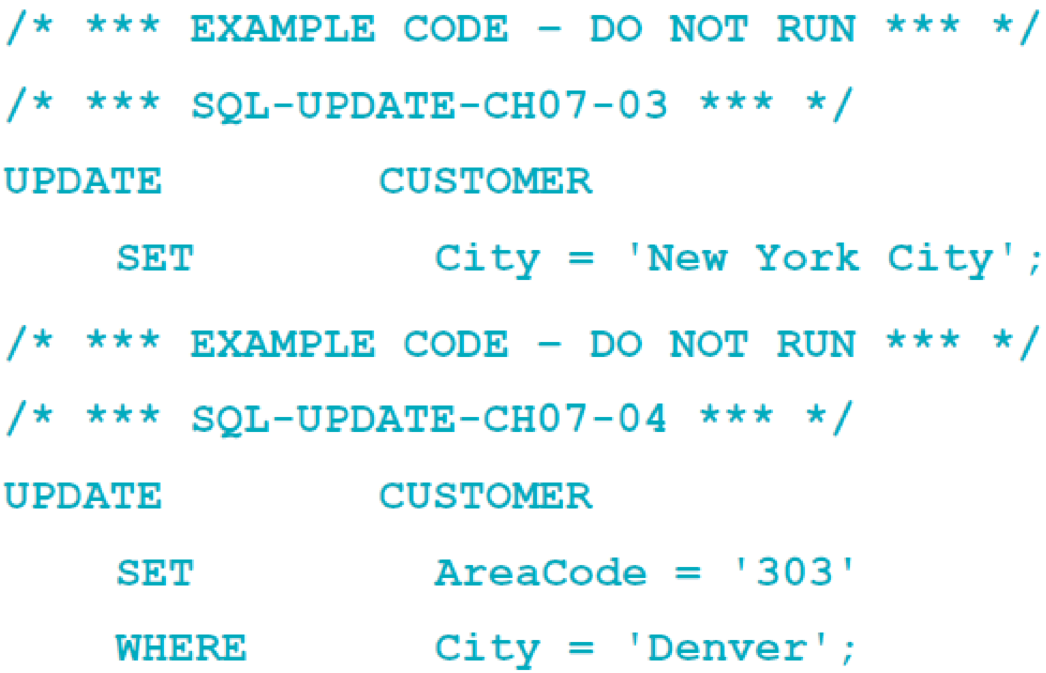
- The **SQL UPDATE statement**:



```
/* *** EXAMPLE CODE – DO NOT RUN *** */
/* *** SQL-UPDATE-CH07-01 *** */
UPDATE        CUSTOMER
    SET         City = 'New York City'
    WHERE       CustomerID = 1000;
/* *** EXAMPLE CODE – DO NOT RUN *** */
/* *** SQL-UPDATE-CH07-02 *** */
UPDATE        CUSTOMER
    SET         City = 'New York City', State = 'NY'
    WHERE       CustomerID = 1000;
```

# SQL DML—UPDATE II

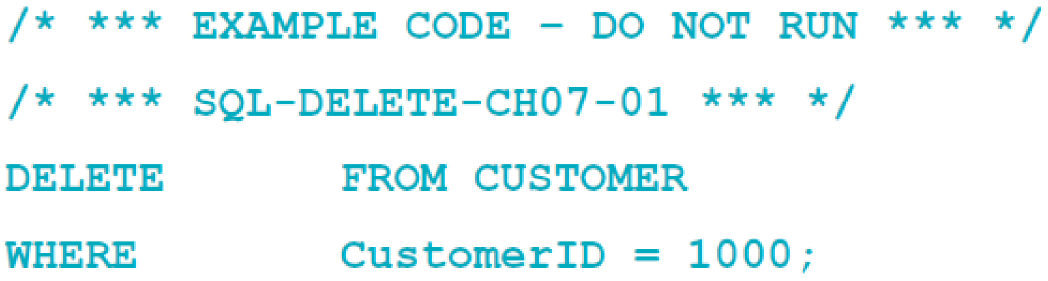
- Bulk UPDATE:



```
/* *** EXAMPLE CODE - DO NOT RUN *** */
/* *** SQL-UPDATE-CH07-03 *** */
UPDATE        CUSTOMER
    SET         City = 'New York City';
/* *** EXAMPLE CODE - DO NOT RUN *** */
/* *** SQL-UPDATE-CH07-04 *** */
UPDATE        CUSTOMER
    SET         AreaCode = '303'
    WHERE       City = 'Denver';
```

# SQL DML—DELETE

- SQL DELETE statement:



```
/* *** EXAMPLE CODE - DO NOT RUN *** */
/* *** SQL-DELETE-CH07-01 *** */
DELETE          FROM CUSTOMER
WHERE           CustomerID = 1000;
```

- Note: If you omit the WHERE clause, you will delete every row in the table.

David Kroenke and David Auer

# Database Processing

Fundamentals, Design, and Implementation

(14th Edition)

## End of Presentation:
## Chapter Seven