

# CSI 2132 Lab#2

- Basic SQL Programming

Presented by: Rana Khalil, 22 Jan 2018

# Updates / Comments

- Lab attendance is mandatory starting this week
- If we don't finish the lab during the lab time, you need to complete it on your own time – labs might build on each other.
- Updated slides are not posted on Brightspace. You can access them through my github account: <https://github.com/rkhal101/CSI2132-Databases-I>

# Outline

- **Review syntax of:**
  - Creating a new Schema
  - CREATE TABLE
  - INSERT
  - SELECT
  - UPDATE
  - DELETE
- **Exercises to cover:**
  - Create a new schema and set it as default
  - Creating tables from ER Diagrams
  - Inserting data to tables
  - Querying the database
  - Updating specific data
  - Deleting specific data

# Create a new Schema and set it to default

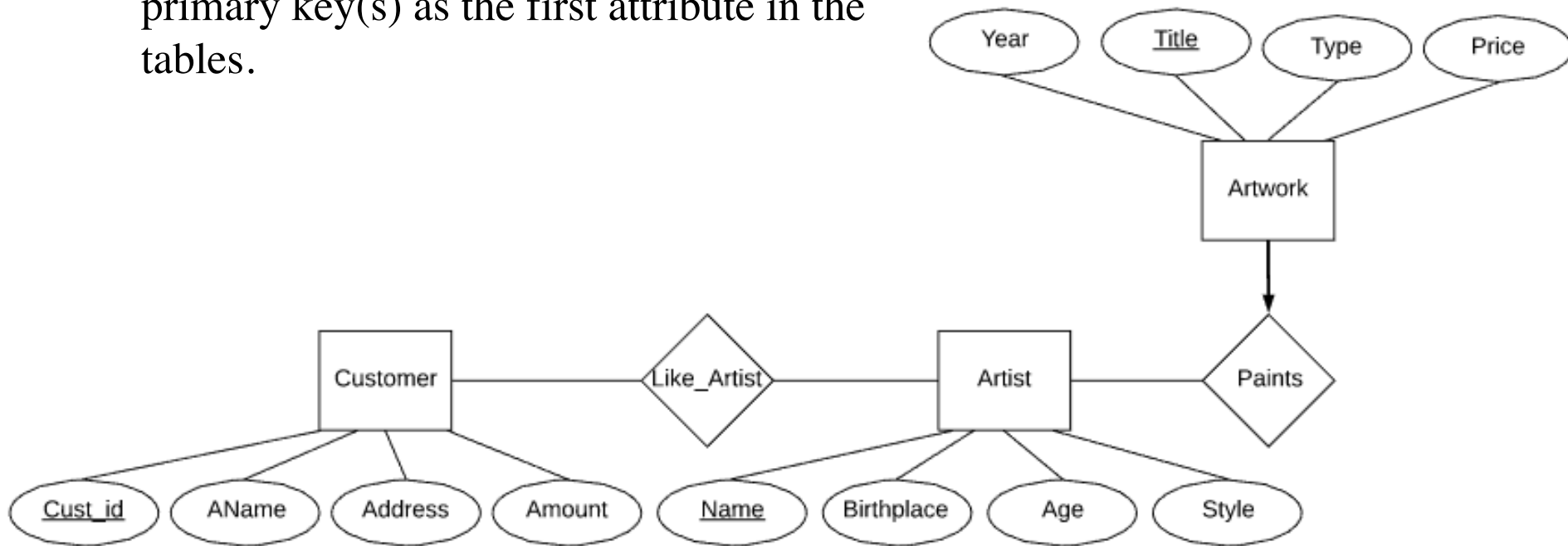
- Right click on **Schemas** in tree control > click on **Create** > click on **Schema**
- In the **Create-Schema** window, in **General** tab name the new schema “laboratories” and owner “your username”.
- In order to use this newly created schema as the default one for the queries instead of the “public” schema, execute this statement using the Query Tool:
  - **SET search\_path = laboratories;**Note that this will be applied for the current session ONLY.

# ER Diagram

- Tables Artist, Artwork, Customer, LikeArtist (many to many relationship)
- Remember the primary key and foreign key constraints.
- A good convention is declaring the primary key(s) as the first attribute in the tables.

Some Useful Data Types:

- VARCHAR(n)
- NUMERIC(n,m)
- INTEGER



# SQL CREATE TABLE Statement

- Syntax

```
CREATE TABLE TableName  
(  
    column1 datatype1,  
    column2 datatype2,  
    ...  
    columnN datatypeN,  
    constraint1, constraint2, ... , constraintM  
);
```

- Example

```
CREATE TABLE Artist  
(  
    AName VARCHAR(20),  
    Birthplace VARCHAR(20),  
    Style VARCHAR(20),  
    Age INTEGER,  
    PRIMARY KEY (AName)  
);
```

# Tables to be Created

Artist		
AName	varchar(20)	primary key
Birthplace	varchar(20)	
Age	integer	
Style	varchar(20)	

Customer		
Cust_id	integer	primary key
Name	varchar(20)	
Address	varchar(20)	
Amount	numeric(8,2)	

Artwork		
Title	varchar(20)	primary key
Year	integer	
Type	varchar(20)	
Price	numeric(8,2)	
AName	varchar(20)	foreign key

Like_Artist		
Cust_id	integer	foreign key
AName	varchar(20)	foreign key
(Cust_id, AName)		primary key

# The Code for all the Tables

```
CREATE TABLE Artist
```

```
(  
    AName VARCHAR(20),  
    Birthplace VARCHAR(20),  
    Style VARCHAR(20),  
    Age INTEGER,  
    PRIMARY KEY (AName)  
);
```

```
CREATE TABLE Artwork
```

```
(  
    Title VARCHAR(20),  
    Year INTEGER,  
    Type VARCHAR(20),  
    AName VARCHAR(20),  
    Price NUMERIC(8,2),  
    PRIMARY KEY (Title),  
    FOREIGN KEY (AName) REFERENCES Artist  
);
```

```
CREATE TABLE Customer
```

```
(  
    Cust_id INTEGER,  
    Name VARCHAR(20),  
    Address VARCHAR(20),  
    Amount NUMERIC(8,2),  
    PRIMARY KEY (Cust_id)  
);
```

```
CREATE TABLE LikeArtist
```

```
(  
    Cust_id INTEGER,  
    AName VARCHAR(20),  
    PRIMARY KEY(AName, Cust_id),  
    FOREIGN KEY (AName) REFERENCES Artist,  
    FOREIGN KEY (Cust_id) REFERENCES Customer  
);
```



# SQL INSERT INTO Statment

- Syntax  

```
INSERT INTO TableName(column1, ... ,columnN) VALUES (value1, ... ,valueN);
```
- Example  

```
INSERT INTO Artist(AName, Birthplace, Style, Age) VALUES ('Caravaggio', 'Milan', 'Baroque', '59');
```
- Character values are quoted by ' ', and numerical values are unquoted when inserting.
- Several inserts can be done consecutively in the Query Tool.

# Values to be Inserted

- Customer(Cust\_id, Name, Address, Amount) Table
  - (1, 'John', 'Ottawa', 8.5)
  - (2, 'Amy', 'Orleans', 9.0)
  - (3, 'Peter', 'Gatineau', 6.3)
- Artist(AName, Birthplace, Style, Age)
  - ('Caravaggio', 'Milan', 'Baroque', '59')
  - ('Smith', 'Ottawa', 'Modern', '33')
  - ('Picasso', 'Malaga', 'Cubism', '40')
- Artwork(Title, Year, Type, Price, AName)
  - ('Blue', 2000, 'Modern', 10000.00, 'Smith')
  - ('The Cardsharps', 1594, 'Baroque', 40000.00, 'Caravaggio')

# The Code for Insertion

```
-- Insert into Customer table
INSERT INTO Customer(Cust_id, Name, Address, Amount) VALUES
(1, 'John', 'Ottawa', 8.5);
INSERT INTO Customer(Cust_id, Name, Address, Amount) VALUES (2, 'Amy',
'Orleans', 9.0);
INSERT INTO Customer(Cust_id, Name, Address, Amount) VALUES (3, 'Peter',
'Gatineau', 6.3);

-- Insert into Artist table
INSERT INTO Artist(AName, Birthplace, Style, Age) VALUES ('Caravaggio',
'Milan', 'Baroque', '59');
INSERT INTO Artist(AName, Birthplace, Style, Age) VALUES ('Smith', 'Ottawa',
'Modern', '33');
INSERT INTO Artist(AName, Birthplace, Style, Age) VALUES ('Picasso',
'Malaga', 'Cubism', '40');

-- Insert into Artwork table
INSERT INTO Artwork(Title, Year, Type, Price, AName) VALUES ('Blue', 2000,
'Modern', 10000.00, 'Smith');
INSERT INTO Artwork(Title, Year, Type, Price, AName) VALUES ('The
Cardsharps', 1594, 'Baroque', 40000.00, 'Caravaggio');
```

# SQL SELECT STATEMENT

- Syntax

```
SELECT column1, column2, ... ,columnN  
FROM table_name;  
WHERE condition;
```

- Example

```
SELECT Style  
FROM Artist  
WHERE AName='Smith';
```

# Exercise: Create the following Queries

- List all artists that are born in Ottawa
- List the titles and prices of all artworks painted in 2000.

# Answer: Create the following Queries

- List all artists that are born in Ottawa  

```
SELECT *  
FROM Artist  
WHERE Birthplace='Ottawa';
```
- List the titles and prices of all artworks painted in 2000.  

```
SELECT Title, Price  
FROM Artwork  
WHERE Year=2000;
```

# SQL UPDATE Statement

- We can also modify certain data satisfying a condition from a table with UPDATE command. Condition is the same as WHERE clause of SELECT query. If you omit the WHERE clause, **all records will be updated permanently**.
- Syntax  

```
UPDATE table_name  
SET column1 = value1, column2 = value2, ... , columnN = valueN  
WHERE condition;
```
- Example  

```
UPDATE Customer  
SET Name = 'Bruce'  
WHERE Cust_id = 1;
```

# Exercise: Update the Following Data

- Update Customer Name John to Bruce.
- Update the Amount value for all the Customers in the Database to be 9.8 and the address to be Gatineau.



# Answer: Create the following Queries

- Update Customer Name John to Bruce.  
`UPDATE Customer  
SET Name='Bruce'  
WHERE Name='John';`
- Update the Amount value for all the Customers in the Database to be 9.8 and the address to be Gatineau.  
`UPDATE Customer  
SET Amount=9.8, Address='Gatineau';`

# SQL DELETE STATEMENT

- We can delete certain rows satisfying a condition from a table with DELETE command. Condition is the same as WHERE clause of SELECT query. If you omit the WHERE clause, **all records will be deleted permanently**.
- Syntax  

```
DELETE FROM table_name  
WHERE condition;
```
- Example  

```
DELETE FROM Customer  
WHERE Cust_id=1;
```

## Exercise: Delete the Following Rows

- Remove Customer Amy from our Database.
- Remove all the remaining Customers from the database.
- Suppose the artist 'Smith' moved to another gallery, and we have to remove him from our database. (Note that Artwork table has a foreign key to Artist table)

# Answer: Delete the Following Rows

- Remove Customer Amy from our Database.  
`DELETE FROM Customer  
WHERE Name= 'Amy' ;`
- Remove all the remaining Customers from the database.  
`DELETE FROM Customer;`
- Suppose the artist 'Smith' moved to another gallery, and we have to remove him from our database. (Note that Artwork table has a foreign key to Artist table)
  - First delete entry from Artwork table  
`DELETE FROM Artwork WHERE AName='Smith';`
  - Second delete entry from Artist table  
`DELETE FROM Artist WHERE AName='Smith';`

# For Detailed Information

- About SQL Syntax
  - <http://www.faqs.org/docs/ppbook/c22759.htm>