

4.26.0.RELEASE - JDBCCon/src/main/java/JDBCDemo/JDBCCon/CustomerInsert.java - Spring Tool Suite 4

File Edit Search Project Run Window Help



JDBCCon/pom.xml \*CustomerInsert.java

```
7 import java.util.Scanner;
8
9 public class CustomerInsert {
10     // Database connection details
11     private static final String JDBC_URL = "jdbc:mysql://localhost:3306/CustomerDB";
12     private static final String USERNAME = "root"; // Change if needed
13     private static final String PASSWORD = "aarathy@123"; // Change if you set a password
14
15     public static void main(String[] args) {
16         Scanner scanner = new Scanner(System.in);
17
18         // Taking user input
19         System.out.print("Enter Customer Name: ");
20         String name = scanner.nextLine();
21
22         System.out.print("Enter Customer Email: ");
23         String email = scanner.nextLine();
24
25         System.out.print("Enter Customer Phone: ");
26         String phone = scanner.nextLine();
27     }
}
```

E-1.8]

/annotations  
sources/test-annotations

Problems Javadoc Declaration Console

```
<terminated> CustomerInsert [Java Application] C:\Users\DELL\Desktop\sts-4.26.0.RELEASE\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win
Enter Customer Name: aarthy
Enter Customer Email: aarthy@gmail.com
Enter Customer Phone: 8765432108
Enter Customer City: chengalpattu
Customer data inserted successfully!
```

names to match (incl. \* an

ance MySQL - x

Query Database Server Tools Scripting Help

SQL File 5\* x

Limit to 1000 rows

```
2 • USE Customerdb;
3 • CREATE TABLE customers (
4     id INT AUTO_INCREMENT PRIMARY KEY,
5     name VARCHAR(100),
6     email VARCHAR(100) UNIQUE,
```

Result Grid

id	name	email	phone	city
1	aarthy	aarthy@gmail.com	8765432108	chengalpattu
NULL	NULL	NULL	NULL	NULL

customers 12 customers 15 customers 18 customers 21 customers 24 customers 27 Result 28 Result Apply Revert Context Help Snippets

Output

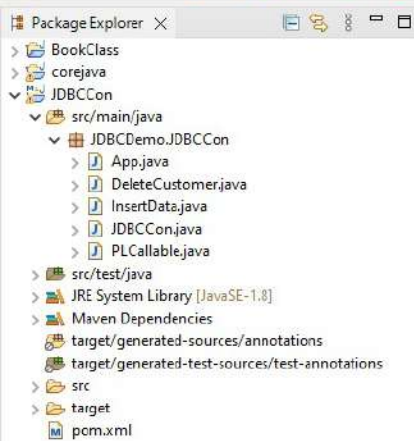
Action Output

#	Time	Action	Message	Duration / Fetch
56	12:52:06	CREATE TABLE customers ( id INT AUTO_INCREMENT PRIMARY KEY, name VARC...	Error Code: 1050. Table 'customers' already exists	0.000 sec
57	12:52:06	SHOW TABLES	1 row(s) returned	0.015 sec / 0.000 sec
58	12:52:06	DESC customers	5 row(s) returned	0.000 sec / 0.000 sec
59	12:52:06	SELECT * FROM customers LIMIT 0, 1000	8 row(s) returned	0.000 sec / 0.000 sec

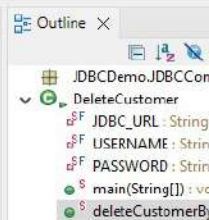
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

workspace-spring-tool-suite-4-4.26.0.RELEASE - JDBCCon/src/main/java/JDBCDemo/JDBCCon/DeleteCustomer.java - Spring Tool Suite 4

File Edit Source Refactor Navigate Search Project Run Window Help

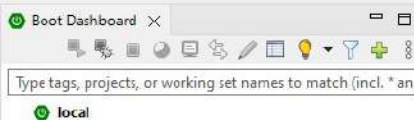


```
22     deleteCustomerId(customerId);
23
24     scanner.close();
25 }
26
27 public static void deleteCustomerId(int customerId) {
28     String deleteQuery = "DELETE FROM customers WHERE id = ?";
29
30     try (Connection conn = DriverManager.getConnection(JDBC_URL, USERNAME, PASSWORD);
31          PreparedStatement pstmt = conn.prepareStatement(deleteQuery)) {
32
33         // Set parameter value
34         pstmt.setInt(1, customerId);
35
36         // Execute delete query
37         int rowsDeleted = pstmt.executeUpdate();
38         if (rowsDeleted > 0) {
39             System.out.println("Customer with ID " + customerId + " deleted successfully!");
40         } else {
41             System.out.println("Customer with ID " + customerId + " not found.");
42         }
43     }
```



Problems @ Javadoc Declaration Console

```
<terminated> DeleteCustomer [Java Application] C:\Users\DELL\Desktop\sts-4.26.0.RELEASE\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.0.4.v20240802-155
Enter Customer ID to delete: 2
Customer with ID 2 deleted successfully!
```



Edit View Query Database Server Tools Scripting Help



SQL File 6

HEMAS

Filter objects

amitdb

customer

Tables

Views

Functions

Stored Procedures

Triggers

Schema: customer

Tables

Views

Functions

Stored Procedures

Triggers

Schema: customer

Tables

Views

Functions

Stored Procedures

Triggers

Schema: customer

Tables

Views

Functions

Stored Procedures

Triggers

Schema: customer

Tables

Views

Functions

Stored Procedures

Triggers

Schema: customer

Tables

Views

Functions

Stored Procedures

Triggers

Schema: customer

Tables

Views

Functions

Stored Procedures

Triggers

Schema: customer

Tables

SQL File 6

Limit to 1000 rows

1 CREATE DATABASE IF NOT EXISTS CustomerDB;

2 USE CustomerDB;

3

4 CREATE TABLE IF NOT EXISTS customers (

5 id INT AUTO INCREMENT PRIMARY KEY,

6 name VARCHAR(100),

7 email VARCHAR(100),

8 phone VARCHAR(15),

9 city VARCHAR(50);

10 );

11 INSERT INTO customers (name, email, phone, city)

12 VALUES

13 (1, 'John Doe', 'john.doe@example.com', '9876543210', 'New York');

14 (2, 'Alice Johnson', 'alice.johnson@example.com', '0987654321', 'Los Angeles');

15 (3, 'Bob Smith', 'bob.smith@example.com', '1234567890', 'Chicago');

16 (4, 'Carol Davis', 'carol.davis@example.com', '2345678901', 'San Francisco');

17 (5, 'David Wilson', 'david.wilson@example.com', '3456789012', 'Austin');

18 (6, 'Eve Brown', 'eve.brown@example.com', '4567890123', 'Boston');

19 (7, 'Frank Miller', 'frank.miller@example.com', '5678901234', 'Denver');

20 (8, 'Grace Lee', 'grace.lee@example.com', '6789012345', 'Houston');

21 (9, 'Henry King', 'henry.king@example.com', '7890123456', 'Indianapolis');

22 (10, 'Ivy White', 'ivy.white@example.com', '8901234567', 'Jacksonville');

23 (11, 'Jack Black', 'jack.black@example.com', '9012345678', 'Kansas City');

24 (12, 'Karen Green', 'karen.green@example.com', '0123456789', 'Las Vegas');

25 (13, 'Leo Gray', 'leo.gray@example.com', '1234567890', 'Long Beach');

26 (14, 'Mia Blue', 'mia.blue@example.com', '2345678901', 'Los Angeles');

27 (15, 'Noah Red', 'noah.red@example.com', '3456789012', 'Louisville');

28 (16, 'Olivia Purple', 'olivia.purple@example.com', '4567890123', 'Madison');

29 (17, 'Peter Yellow', 'peter.yellow@example.com', '5678901234', 'Miami');

30 (18, 'Quinn Orange', 'quinn.orange@example.com', '6789012345', 'Milwaukee');

31 (19, 'Ryan Silver', 'ryan.silver@example.com', '7890123456', 'Minneapolis');

32 (20, 'Sara Bronze', 'sara.bronze@example.com', '8901234567', 'New Orleans');

33 (21, 'Tina Copper', 'tina.copper@example.com', '9012345678', 'New York');

34 (22, 'Uma Gold', 'uma.gold@example.com', '0123456789', 'New York');

35 (23, 'Victor Platinum', 'victor.platinum@example.com', '1234567890', 'New York');

36 (24, 'Wendy Diamond', 'wendy.diamond@example.com', '2345678901', 'New York');

37 (25, 'Xavier Ruby', 'xavier.ruby@example.com', '3456789012', 'New York');

38 (26, 'Yara Sapphire', 'yara.sapphire@example.com', '4567890123', 'New York');

39 (27, 'Zoe Emerald', 'zoe.emerald@example.com', '5678901234', 'New York');

40 (28, 'Adam Onyx', 'adam.onyx@example.com', '6789012345', 'New York');

41 (29, 'Bella Obsidian', 'bella.obsidian@example.com', '7890123456', 'New York');

42 (30, 'Caleb Jade', 'caleb.jade@example.com', '8901234567', 'New York');

43 (31, 'Diana Garnet', 'diana.garnet@example.com', '9012345678', 'New York');

44 (32, 'Ethan Amethyst', 'ethan.amethyst@example.com', '0123456789', 'New York');

45 (33, 'Fiona Topaz', 'fiona.topaz@example.com', '1234567890', 'New York');

46 (34, 'Gavin Peridot', 'gavin.peridot@example.com', '2345678901', 'New York');

47 (35, 'Hannah Aquamarine', 'hannah.aquamarine@example.com', '3456789012', 'New York');

SQLAdditions

Automatic

disabled. U

manually get

caret pos

auto

Automatic

disabled. U

manually get

caret pos

auto

Automatic

disabled. U

manually get

caret pos

auto

Automatic

disabled. U

manually get

caret pos

auto

Automatic

disabled. U

manually get

caret pos

auto

Automatic

disabled. U

manually get

caret pos

auto

Automatic

disabled. U

manually get

caret pos

auto

Automatic

disabled. U

manually get

caret pos

auto

Automatic

disabled. U

manually get

caret pos

auto

Automatic

disabled. U

manually get