

Name: Abhishek Kanoujia

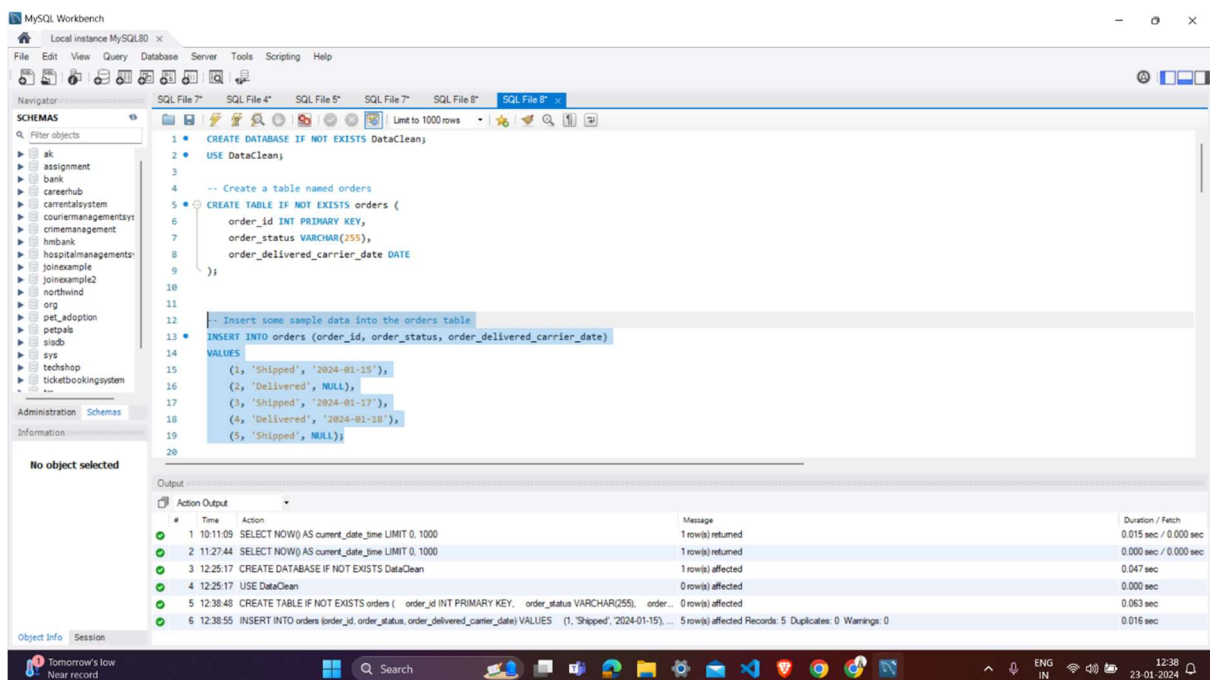
DATA ENGINEERING BATCH 1

DAY 5 ASSIGNMENT

Table of Contents

1. Check for Missing Values
2. Check for Duplicates
3. Standardizing and Transforming Data
4. Updating Data / Column Data Types

All the queries related to this mentioned in the screen shorts



MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- ak
- assignment
- bank
- careerhub
- careerhubsystem
- couriermanagementsys
- crimemanagement
- hmbank
- hospitalmanagements
- joinexample
- joinexample2
- northwind
- org
- pet_adoption
- petpals
- slsdb
- sys
- techshop
- ticketbookingsystem

Administration Schemas

Information

No object selected

SQL File 7 SQL File 4 SQL File 5 SQL File 7 SQL File 8 SQL File 9

Limit to 1000 rows

22 -- 1. Check for Missing Values

23 Execute the selected portion of the script or everything, if there is no selection

24

25 -- Check for missing values in the order_delivered_carrier_date column

26 SELECT 'Missing Values Check:', order_id, order_status, order_delivered_carrier_date

27 FROM orders

28 WHERE order_delivered_carrier_date IS NULL;

29

30 -- Replace missing values in order_delivered_carrier_date with 'Unknown'

31 SELECT 'COALESCE to Replace Missing Values:', order_id, order_status, COALESCE(order_delivered_carrier_date, 'Unknown') AS filled_column

32 FROM orders;

33

34

35 -- 2. Check for Duplicate Rows

Result Grid

Missing Values Check:	order_id	order_status	order_delivered_carrier_date
Missing Values Check:	2	Delivered	NULL
Missing Values Check:	5	Shipped	NULL

Output

#	Time	Action	Message	Duration / Fetch
2	11:27:44	SELECT NOW() AS current_date_time LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
3	12:25:17	CREATE DATABASE IF NOT EXISTS DataClean	1 row(s) affected	0.047 sec
4	12:25:17	USE DataClean	0 row(s) affected	0.000 sec
5	12:38:48	CREATE TABLE IF NOT EXISTS orders (order_id INT PRIMARY KEY, order_status VARCHAR(255), ord...	0 row(s) affected	0.063 sec
6	12:38:55	INSERT INTO orders (order_id, order_status, order_delivered_carrier_date) VALUES (1, 'Shipped', '2024-01-15'...	5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.016 sec
7	12:39:28	SELECT Missing Values Check:, order_id, order_status, order_delivered_carrier_date FROM orders WHERE ord...	2 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- ak
- assignment
- bank
- careerhub
- careerhubsystem
- couriermanagementsys
- crimemanagement
- hmbank
- hospitalmanagements
- joinexample
- joinexample2
- northwind
- org
- pet_adoption
- petpals
- slsdb
- sys
- techshop
- ticketbookingsystem

Administration Schemas

Information

No object selected

SQL File 7 SQL File 4 SQL File 5 SQL File 7 SQL File 8 SQL File 9

Limit to 1000 rows

25 -- Check for missing values in the order_delivered_carrier_date column

26 SELECT 'Missing Values Check:', order_id, order_status, order_delivered_carrier_date

27 FROM orders

28 WHERE order_delivered_carrier_date IS NULL;

29

30 -- Replace missing values in order_delivered_carrier_date with 'Unknown'

31 SELECT 'COALESCE to Replace Missing Values:', order_id, order_status, COALESCE(order_delivered_carrier_date, 'Unknown') AS filled_column

32 FROM orders;

33

34

35 -- 2. Check for Duplicate Rows

Result Grid

COALESCE to Replace Missing Values:	order_id	order_status	filled_column
COALESCE to Replace Missing Values:	1	Shipped	2024-01-15
COALESCE to Replace Missing Values:	2	Delivered	Unknown
COALESCE to Replace Missing Values:	3	Shipped	2024-01-17
COALESCE to Replace Missing Values:	4	Delivered	2024-01-18
COALESCE to Replace Missing Values:	5	Shipped	Unknown

Output

#	Time	Action	Message	Duration / Fetch
3	12:25:17	CREATE DATABASE IF NOT EXISTS DataClean	1 row(s) affected	0.047 sec
4	12:25:17	USE DataClean	0 row(s) affected	0.000 sec
5	12:38:48	CREATE TABLE IF NOT EXISTS orders (order_id INT PRIMARY KEY, order_status VARCHAR(255), ord...	0 row(s) affected	0.063 sec
6	12:38:55	INSERT INTO orders (order_id, order_status, order_delivered_carrier_date) VALUES (1, 'Shipped', '2024-01-15'...	5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.016 sec
7	12:39:28	SELECT Missing Values Check:, order_id, order_status, order_delivered_carrier_date FROM orders WHERE ord...	2 row(s) returned	0.000 sec / 0.000 sec
8	12:39:38	SELECT COALESCE to Replace Missing Values:, order_id, order_status, COALESCE(order_delivered_carrier_da...	5 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

Schemas

Filter objects

- ak
- assignment
- bank
- careerhub
- carrentalsystem
- couriermanagementsys
- crimemanagement
- hmbank
- hospitalmanagements
- joinexample
- joinexample2
- northwind
- org
- pet_adoption
- petals
- slsdb
- sys
- techshop
- ticketbookingsystem

Administration Schemas

Information

No object selected

SQL File 7* SQL File 4* SQL File 5* SQL File 7* SQL File 8* SQL File 9*

Limit to 1000 rows

```

34
35 -- 2. Check for Duplicate Rows
36
37
38 -- Check for duplicate rows based on order_id and order_status
39 • SELECT 'Duplicate Rows Check:', order_id, order_status, COUNT(*) AS count
40 FROM orders
41 GROUP BY order_id, order_status
42 HAVING COUNT(*) > 1;
43
44

```

Result Grid

Duplicate Rows Check:	order_id	order_status	count
-----------------------	----------	--------------	-------

Output

#	Time	Action	Message	Duration / Fetch
4	12:25:17	USE DataClean	0 row(s) affected	0.000 sec
5	12:38:48	CREATE TABLE IF NOT EXISTS orders (order_id INT PRIMARY KEY, order_status VARCHAR(255), ord...	0 row(s) affected	0.063 sec
6	12:38:55	INSERT INTO orders (order_id, order_status, order_delivered_carrier_date) VALUES (1, 'Shipped', '2024-01-15'...	5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0	0.016 sec
7	12:39:28	SELECT Missing Values Check: order_id, order_status, order_delivered_carrier_date FROM orders WHERE ord...	2 row(s) returned	0.000 sec / 0.000 sec
8	12:39:38	SELECT COALESCE to Replace Missing Values: order_id, order_status, COALESCE(order_delivered_carrier_da...	5 row(s) returned	0.000 sec / 0.000 sec
9	12:40:10	SELECT Duplicate Rows Check: order_id, order_status, COUNT(*) AS count FROM orders GROUP BY order_id...	0 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

Schemas

Filter objects

- ak
- assignment
- bank
- careerhub
- carrentalsystem
- couriermanagementsys
- crimemanagement
- hmbank
- hospitalmanagements
- joinexample
- joinexample2
- northwind
- org
- pet_adoption
- petals
- slsdb
- sys
- techshop
- ticketbookingsystem

Administration Schemas

Information

No object selected

SQL File 7* SQL File 4* SQL File 5* SQL File 7* SQL File 8* SQL File 9*

Limit to 1000 rows

```

45 -- 3. Standardizing and Transforming Data
46
47 -- Standardize date: Convert order_status to uppercase
48 • UPDATE orders
49 SET order_status = UPPER(order_status);
50
51 -- Transform data: Change 'Shipped' to 'In Transit'
52 • UPDATE orders
53 SET order_status = 'In Transit'
54 WHERE order_status = 'Shipped';
55
56 -- Display the transformed data
57 • SELECT 'Transformed Data:', order_id, order_status, order_delivered_carrier_date
58 FROM orders;
59
60 -- Convert product_category_name to uppercase
61 • SELECT 'UPPER to Standardize Data:', product_id, UPPER(product_category_name) AS uppercase_column
62 FROM products;
63
64 -- Replace 'St.' with 'Street' in the address field

```

Output

#	Time	Action	Message	Duration / Fetch
8	12:39:38	SELECT COALESCE to Replace Missing Values: order_id, order_status, COALESCE(order_delivered_carrier_da...	5 row(s) returned	0.000 sec / 0.000 sec
9	12:40:10	SELECT Duplicate Rows Check: order_id, order_status, COUNT(*) AS count FROM orders GROUP BY order_id...	0 row(s) returned	0.000 sec / 0.000 sec
10	12:40:29	UPDATE orders SET order_status = UPPER(order_status)	Error Code: 1175. You are using safe update mode and you tried to update a table without a WHERE that uses a ...	0.000 sec
11	12:41:06	UPDATE orders SET order_status = 'In Transit' WHERE order_status = 'Shipped'	Error Code: 1175. You are using safe update mode and you tried to update a table without a WHERE that uses a ...	0.000 sec
12	12:41:19	SELECT 'Transformed Data:', FROM orders	Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds to your MySQL serve...	0.000 sec
13	12:43:54	SELECT 'Transformed Data:', order_id, order_status, order_delivered_carrier_date FROM orders LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- ak
- assignment
- bank
- careerhub
- carrentalsystem
- couriermanagement
- crimemanagement
- hmbank
- hospitalmanagement
- joinexample
- joinexample2
- northwind
- org
- pet_adoption
- petals
- slsdb
- sys
- techshop
- ticketbooking

Administration Schemas

Information

SQL File 7 SQL File 4 SQL File 5 SQL File 7 SQL File 8 SQL File 8

Limit to 1000 rows

51 -- Transform data: Change 'Shipped' to 'In Transit'

52 UPDATE orders

53 SET order_status = 'In Transit'

54 WHERE order_status = 'Shipped';

55

56 -- Display the transformed data

57 SELECT 'Transformed Data:', order_id, order_status, order_delivered_carrier_date

58 FROM orders;

59

60 -- Convert product_category_name to uppercase

61 SELECT 'UPPER to Standardize Data:', product_id, UPPER(product_category_name) AS uppercase_column

Result Grid

Transformed Data:	order_id	order_status	order_delivered_carrier_date
Transformed Data: 1	1	Shipped	2024-01-15
Transformed Data: 2	2	Delivered	2024-01-16
Transformed Data: 3	3	Shipped	2024-01-17
Transformed Data: 4	4	Delivered	2024-01-18
Transformed Data: 5	5	Shipped	2024-01-19

No object selected

Result 5

Output

#	Time	Action	Message	Duration / Fetch
9	12:40:10	SELECT Duplicate Rows Check: order_id, order_status, COUNT(*) AS count FROM orders GROUP BY order_id	0 row(s) returned	0.000 sec / 0.000 sec
10	12:40:29	UPDATE orders SET order_status = UPPER(order_status)	Error Code: 1175. You are using safe update mode and you tried to update a table without a WHERE that uses a ...	0.000 sec
11	12:41:06	UPDATE orders SET order_status = 'In Transit' WHERE order_status = 'Shipped'	Error Code: 1175. You are using safe update mode and you tried to update a table without a WHERE that uses a ...	0.000 sec
12	12:41:19	SELECT 'Transformed Data:', order_id, order_status, order_delivered_carrier_date FROM orders	Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds to your MySQL serve...	0.000 sec
13	12:43:54	SELECT 'Transformed Data:', order_id, order_status, order_delivered_carrier_date FROM orders LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
14	12:44:33	SELECT 'Transformed Data:', order_id, order_status, order_delivered_carrier_date FROM orders LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- ak
- assignment
- bank
- careerhub
- carrentalsystem
- couriermanagement
- crimemanagement
- hmbank
- hospitalmanagement
- joinexample
- joinexample2
- northwind
- org
- pet_adoption
- petals
- slsdb
- sys
- techshop
- ticketbooking

Administration Schemas

Information

SQL File 7 SQL File 4 SQL File 5 SQL File 7 SQL File 8 SQL File 8

Limit to 1000 rows

136 UPDATE orders

137 SET order_status = 'In Transit'

138 WHERE order_status = 'Shipped';

139

140 -- Display the transformed data

141 SELECT 'Transformed Data:', order_id, order_status, order_delivered_carrier_date

142 FROM orders;

143

144 -- Convert product_category_name to uppercase

145 SELECT 'UPPER to Standardize Data:', product_id, UPPER(product_category_name) AS uppercase_column

146 FROM products;

Result Grid

UPPER to Standardize Data:	product_id	uppercase_column
UPPER to Standardize Data: 101	101	ELECTRONICS
UPPER to Standardize Data: 102	102	CLOTHING
UPPER to Standardize Data: 103	103	HOME AND KITCHEN

No object selected

Result 6

Output

#	Time	Action	Message	Duration / Fetch
24	14:08:46	INSERT INTO sales (sale_id, qty, sales_category) VALUES (401, 30, 'Medium Sales'), (402, 10, 'Low Sales')...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
25	14:08:52	CREATE TABLE IF NOT EXISTS customers (customer_id INT PRIMARY KEY, email VARCHAR(255))	0 row(s) affected	0.016 sec
26	14:08:58	INSERT INTO customers (customer_id, email) VALUES (501, 'customer1@example.com'), (502, 'customer2...)	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
27	14:09:04	CREATE TABLE IF NOT EXISTS stores (store_id INT PRIMARY KEY, location VARCHAR(255))	0 row(s) affected	0.016 sec
28	14:09:10	INSERT INTO stores (store_id, location) VALUES (601, 'New York'), (602, 'Los Angeles'), (603, 'Chicago')	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
29	14:09:40	SELECT 'UPPER to Standardize Data:', product_id, UPPER(product_category_name) AS uppercase_column FR	3 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- ak
- assignment
- bank
- careerhub
- carrentalsystem
- couriermanagementsys
- crimemanagement
- hmbank
- hospitalmanagements
- joinexample
- joinexample2
- northwind
- org
- pet_adoption
- petals
- sisdb
- sys
- techshop
- ticketbookingsystem

Administration Schemas

Information

No object selected

SQL File 7 SQL File 4 SQL File 5 SQL File 7 SQL File 8 SQL File 8

Limit to 1000 rows

```
FROM orders;
142
143
144 -- Convert product_category_name to uppercase
145 SELECT 'UPPER to Standardize Data:', product_id, UPPER(product_category_name) AS uppercase_column
146 FROM products;
147
148 -- Replace 'St.' with 'Street' in the address field
149 SELECT 'REPLACE to Transform Data:', address, REPLACE(address, 'St.', 'Street') AS updated_address
150 FROM authors;
151
152 -- Select the first 10 characters of all book titles
```

Result Grid

REPLACE to Transform Data:	address	updated_address
REPLACE to Transform Data:	123 Main St	123 Main St
REPLACE to Transform Data:	456 Oak St	456 Oak St
REPLACE to Transform Data:	789 Maple St	789 Maple St

Output

Action Output

#	Time	Action	Message	Duration / Fetch
25	14:08:52	CREATE TABLE IF NOT EXISTS customers (customer_id INT PRIMARY KEY, email VARCHAR(255))	0 row(s) affected	0.016 sec
26	14:08:58	INSERT INTO customers (customer_id, email) VALUES (501, 'customer1@example.com'), (502, 'customer2...)	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
27	14:09:04	CREATE TABLE IF NOT EXISTS stores (store_id INT PRIMARY KEY, location VARCHAR(255))	0 row(s) affected	0.016 sec
28	14:09:10	INSERT INTO stores (store_id, location) VALUES (601, 'New York'), (602, 'Los Angeles'), (603, 'Chicago')	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
29	14:09:40	SELECT UPPER to Standardize Data:', product_id, UPPER(product_category_name) AS uppercase_column FR...	3 row(s) returned	0.000 sec / 0.000 sec
30	14:09:55	SELECT REPLACE to Transform Data:', address, REPLACE(address, 'St.', 'Street') AS updated_address FROM a...	3 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- ak
- assignment
- bank
- careerhub
- carrentalsystem
- couriermanagementsys
- crimemanagement
- hmbank
- hospitalmanagements
- joinexample
- joinexample2
- northwind
- org
- pet_adoption
- petals
- sisdb
- sys
- techshop
- ticketbookingsystem

Administration Schemas

Information

No object selected

SQL File 7 SQL File 4 SQL File 5 SQL File 7 SQL File 8 SQL File 8

Limit to 1000 rows

```
148 -- Replace 'St.' with 'Street' in the address field
149 SELECT 'REPLACE to Transform Data:', address, REPLACE(address, 'St.', 'Street') AS updated_address
150 FROM authors;
151
152 -- Select the first 10 characters of all book titles
153 SELECT 'SUBSTRING to Transform Data:', SUBSTRING(title, 1, 10) AS short_title
154 FROM titles;
155
156 -- Remove leading and trailing spaces from the address field
157 SELECT 'TRIM to Transform Data:', TRIM(address) AS trimmed_address
158 FROM authors;
```

Result Grid

SUBSTRING to Transform Data:	short_title
SUBSTRING to Transform Data:	Introduci
SUBSTRING to Transform Data:	Data Scen
SUBSTRING to Transform Data:	Web Develo

Output

Action Output

#	Time	Action	Message	Duration / Fetch
26	14:08:58	INSERT INTO customers (customer_id, email) VALUES (501, 'customer1@example.com'), (502, 'customer2...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
27	14:09:04	CREATE TABLE IF NOT EXISTS stores (store_id INT PRIMARY KEY, location VARCHAR(255))	0 row(s) affected	0.016 sec
28	14:09:10	INSERT INTO stores (store_id, location) VALUES (601, 'New York'), (602, 'Los Angeles'), (603, 'Chicago')	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
29	14:09:40	SELECT UPPER to Standardize Data:', product_id, UPPER(product_category_name) AS uppercase_column FR...	3 row(s) returned	0.000 sec / 0.000 sec
30	14:09:55	SELECT REPLACE to Transform Data:', address, REPLACE(address, 'St.', 'Street') AS updated_address FROM a...	3 row(s) returned	0.000 sec / 0.000 sec
31	14:10:17	SELECT 'SUBSTRING to Transform Data:', SUBSTRING(title, 1, 10) AS short_title FROM titles LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- ak
- assignment
- bank
- careerhub
- careerhubsystem
- couriermanagementsys
- crimemanagement
- hmbank
- hospitalmanagements
- joinexample
- joinexample2
- northwind
- org
- pet_adoption
- petals
- slsdb
- sys
- techshop
- ticketbookingsystem

Administration Schemas

Information

No object selected

SQL File 7 SQL File 4 SQL File 5 SQL File 7 SQL File 8 SQL File 8

Limit to 1000 rows

FROM titles;

-- Remove leading and trailing spaces from the address field

SELECT 'TRIM to Transform Data:', TRIM(address) AS trimmed_address

FROM authors;

-- Format ord_date to a specific format

SELECT 'DATE_FORMAT to Transform Data:', ord_num, ord_date, DATE_FORMAT(ord_date, '%d %m %Y') AS formatted_date

FROM sales;

-- Perform conditional data transformations using CASE WHEN

Result Grid

Filter Rows

Export

Wrap Cell Contents

TRIM to Transform Data: trimmed_address

TRIM to Transform Data: 123 Main St

TRIM to Transform Data: 456 Oak St

TRIM to Transform Data: 789 Maple St

Result 9

Output

Action Output

#	Time	Action	Message	Duration / Fetch
27	14:09:04	CREATE TABLE IF NOT EXISTS stores (store_id INT PRIMARY KEY, location VARCHAR(255))	0 row(s) affected	0.016 sec
28	14:09:10	INSERT INTO stores (store_id, location) VALUES (601, 'New York'), (602, 'Los Angeles'), (603, 'Chicago')	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
29	14:09:40	SELECT UPPER to Standardize Data: product_id, UPPER(product_category_name) AS uppercase_column FR...	3 row(s) returned	0.000 sec / 0.000 sec
30	14:09:55	SELECT REPLACE to Transform Data: address, REPLACE(address, 'St', 'Street') AS updated_address FROM a...	3 row(s) returned	0.000 sec / 0.000 sec
31	14:10:17	SELECT SUBSTRING to Transform Data: SUBSTRING(title, 1, 10) AS short_title FROM titles LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
32	14:10:26	SELECT TRIM to Transform Data: TRIM(address) AS trimmed_address FROM authors LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- ak
- assignment
- bank
- careerhub
- careerhubsystem
- couriermanagementsys
- crimemanagement
- hmbank
- hospitalmanagements
- joinexample
- joinexample2
- northwind
- org
- pet_adoption
- petals
- slsdb
- sys
- techshop
- ticketbookingsystem

Administration Schemas

Information

No object selected

SQL File 7 SQL File 4 SQL File 5 SQL File 7 SQL File 8 SQL File 8

Limit to 1000 rows

SELECT 'CASE WHEN for Conditional Transformations:', sales_category, SUM(qty)

FROM (

SELECT

CASE

WHEN qty >= 50 THEN 'high sales'

WHEN qty >= 20 AND qty < 50 THEN 'medium sales'

WHEN qty < 20 THEN 'low sales'

END AS sales_category,

qty

FROM sales

) AS transformed_sales

Result Grid

Filter Rows

Export

Wrap Cell Contents

CASE WHEN for Conditional Transformations: sales_category SUM(qty)

CASE WHEN for Conditional Transformations: medium sales 30

CASE WHEN for Conditional Transformations: low sales 10

CASE WHEN for Conditional Transformations: high sales 60

Result 10

Output

Action Output

#	Time	Action	Message	Duration / Fetch
29	14:09:40	SELECT UPPER to Standardize Data: product_id, UPPER(product_category_name) AS uppercase_column FR...	3 row(s) returned	0.000 sec / 0.000 sec
30	14:09:55	SELECT REPLACE to Transform Data: address, REPLACE(address, 'St', 'Street') AS updated_address FROM a...	3 row(s) returned	0.000 sec / 0.000 sec
31	14:10:17	SELECT SUBSTRING to Transform Data: SUBSTRING(title, 1, 10) AS short_title FROM titles LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
32	14:10:26	SELECT TRIM to Transform Data: TRIM(address) AS trimmed_address FROM authors LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
33	14:10:37	SELECT DATE_FORMAT to Transform Data: ord_num, ord_date, DATE_FORMAT(ord_date, '%d %m %Y') AS f...	Error Code: 1054. Unknown column 'ord_num' in 'field list'	0.000 sec
34	14:13:45	SELECT CASE WHEN for Conditional Transformations: sales_category, SUM(qty) FROM (SELECT CAS...	3 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

Filter objects

SCHMAS

ak
assignment
bank
careerhub
carrentalsystem
couriermanagementsys
crimemanagement
hmbank
hospitalmanagements-
joinexample
joinexample2
northwind
org
pet_adoption
petals
sisdb
sys
techshop
ticketbookingsystem

Administration Schemas
Information

No object selected

SQL File 7* SQL File 4* SQL File 5* SQL File 7* SQL File 8* SQL File 8*

Limit to 1000 rows

```

180
181
182 -- 4. Updating Data / Column Data Types
183
184 -- Update the email of the customer with customer_id 1
185 • UPDATE customers
186 SET email = "john.doe@example.com"
187 WHERE customer_id = 1;
188
189 -- Update location values based on specific conditions
190 • UPDATE stores
191 SET location =
192 CASE
193 WHEN location = 'New York' THEN 'NY'
194 WHEN location = 'Los Angeles' THEN 'LA'
195 WHEN location = 'Chicago' THEN 'CH'
196 END;
197
198 -- Change the data type of the "address" column to "INT"
199 • ALTER TABLE customers
200 MODIFY COLUMN address INT;
201

```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
29	14:09:40	SELECT UPPER to Standardize Data: product_id, UPPER(product_category_name) AS uppercase_column FR...	3 row(s) returned	0.000 sec / 0.000 sec
30	14:09:55	SELECT REPLACE to Transform Data: address, REPLACE(address, 'St', 'Street') AS updated_address FROM a...	3 row(s) returned	0.000 sec / 0.000 sec
31	14:10:17	SELECT SUBSTRING to Transform Data: SUBSTRING(title, 1, 10) AS short_title FROM titles LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
32	14:10:26	SELECT TRIM to Transform Data: TRIM(address) AS trimmed_address FROM authors LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
33	14:10:37	SELECT DATE_FORMAT to Transform Data: ord_num, ord_date, DATE_FORMAT(ord_date, '%d %M %Y') AS f...	Error Code: 1054. Unknown column 'ord_num' in field list	0.000 sec
34	14:13:45	SELECT CASE WHEN for Conditional Transformations: sales_category, SUM(mdy) FROM (SELECT CAS...	3 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

16°C Haze

Search

ENG IN 14:14 23-01-2024

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

Filter objects

SCHMAS

ak
assignment
bank
careerhub
carrentalsystem
couriermanagementsys
crimemanagement
hmbank
hospitalmanagements-
joinexample
joinexample2
northwind
org
pet_adoption
petals
sisdb
sys
techshop
ticketbookingsystem

Administration Schemas
Information

No object selected

SQL File 7* SQL File 4* SQL File 5* SQL File 7* SQL File 8* SQL File 8*

Limit to 1000 rows

```

185 WHERE customer_id = 1;
186
187 -- Update location values based on specific conditions
188 • UPDATE stores
189 SET location =
190 CASE
191 WHEN location = 'New York' THEN 'NY'
192 WHEN location = 'Los Angeles' THEN 'LA'
193 WHEN location = 'Chicago' THEN 'CH'
194 END;
195
196 -- Change the data type of the "address" column to "INT"
197 • ALTER TABLE customers
198 MODIFY COLUMN address INT;
199
200 -- Drop the "address" column from the "customers" table
201 • ALTER TABLE customers
202 DROP COLUMN address;
203

```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
33	14:10:37	SELECT DATE_FORMAT to Transform Data: ord_num, ord_date, DATE_FORMAT(ord_date, '%d %M %Y') AS f...	Error Code: 1054. Unknown column 'ord_num' in field list	0.000 sec
34	14:13:45	SELECT CASE WHEN for Conditional Transformations: sales_category, SUM(mdy) FROM (SELECT CAS...	3 row(s) returned	0.000 sec / 0.000 sec
35	14:14:14	UPDATE customers SET email = john.doe@example.com WHERE customer_id = 1	0 row(s) affected Rows matched: 0 Changed: 0 Warnings: 0	0.000 sec
36	14:14:20	UPDATE stores SET location = CASE WHEN location = 'New York' THEN 'NY' WHEN location = 'Los Ang...	Error Code: 1175. You are using safe update mode and you tried to update a table without a WHERE that uses a ...	0.000 sec
37	14:14:34	ALTER TABLE customers MODIFY COLUMN address INT	Error Code: 1054. Unknown column 'address' in 'customers'	0.016 sec
38	14:14:45	ALTER TABLE customers DROP COLUMN address	Error Code: 1091. Can't DROP 'address'; check that column/key exists	0.000 sec

Object Info Session

16°C Air: Very poor

Search

ENG IN 14:15 23-01-2024

Using Rank Functions

The functions are used to generate sequence numbers for each row for providing a rank based on the specified condition. The functions are provided to rank the records in the table. The following functions are provided for the rank records.

- 1) row_number()
- 2) rank ()
- 3) dense_rank ()
- 4) ntile ()

mycompiler.io/new/sql

SQL

```
1 -- Create a new database
2 -- CREATE DATABASE Rankexample;
3 -- USE Rankexample;
4
5 -- Create a table for the 'result' example
6 CREATE TABLE result (
7   studid INT,
8   course VARCHAR(255),
9   grade INT
10 );
11
12 -- Insert data into the 'result' table
13 INSERT INTO result (studid, course, grade)
14 VALUES (1, 'Math', 85),
15         (2, 'English', 92),
16         (3, 'Science', 78),
17         (4, 'History', 89);
18
19 -- Perform the ROW_NUMBER() query
20 SELECT studid, grade, ROW_NUMBER() OVER (ORDER BY studid DESC) AS rank
21 FROM result;
```

Run Save

Output

```
4|89|1
3|78|2
2|92|3
1|85|4

[Execution complete with exit code 0]
```

Get 10 free Adobe Stock photos. Start downloading amazing royalty-free stock photos today.

Supported languages

mycompiler.io/new/sql

SQL

```
26
27
28 -- Create a table for the 'rate' example
29 CREATE TABLE rate (
30   orderid INT,
31   productname VARCHAR(255),
32   cost INT
33 );
34
35 -- Insert data into the 'rate' table
36 INSERT INTO rate (orderid, productname, cost)
37 VALUES (101, 'ProductA', 50),
38         (102, 'ProductB', 75),
39         (103, 'ProductC', 60),
40         (104, 'ProductD', 90);
41
42
43 -- Perform the RANK() query
44 SELECT orderid, cost, RANK() OVER (ORDER BY orderid) AS rank
45 FROM rate;
46
47
48
49
50 -- Perform the DENSE_RANK() query
51 SELECT orderid, productname, DENSE_RANK() OVER (ORDER BY orderid DESC) AS rank
52 FROM rate;
```

Run Save

Output

```
101|50|1
102|75|2
103|60|3
104|90|4
104|ProductD|1
103|ProductC|2
102|ProductB|3
101|ProductA|4

[Execution complete with exit code 0]
```

Get 10 free Adobe Stock photos. Start downloading amazing royalty-free stock photos today.

Supported languages

mycompiler.io/new/sql

mycompiler.io/new/sql

myCompiler

EnglishRecentLoginSign up

Enter a title...

Ctrl+Enter

SQL

1

1

2

3 -- Create a table for the 'examdata' example

4 CREATE TABLE examdata (

5 subid INT,

6 subname VARCHAR(255),

7 examdate DATE,

8 location VARCHAR(255)

9);

10

11 -- Insert data into the 'examdata' table

12 INSERT INTO examdata (subid, subname, examdate, location)

13 VALUES (1, 'Math', '2022-01-15', 'Room A'),

14 (2, 'English', '2022-01-17', 'Room B'),

15 (3, 'Science', '2022-01-20', 'Room C'),

16 (4, 'History', '2022-01-22', 'Room D');

17

18 -- Perform the NTILE() query

19 SELECT subid, subname, examdate, NTILE(2) OVER (ORDER BY examdate) AS rank

20 FROM examdata

21 WHERE strftime('%Y', examdate) >= '2011';

22

Output

1|Math|2022-01-15|1

2|English|2022-01-17|1

3|Science|2022-01-20|2

4|History|2022-01-22|2

[Execution complete with exit code 0]

11

Adobe Stock

Get 10 free Adobe Stock photos. Start downloading amazing royalty-free stock photos today.

ADS VIA CARBON

8°C Clear

Search

ENG IN

23:23 23-01-2024

SQL Stored Procedures for SQL Server

A stored procedure is a prepared SQL code that you can save, so the code can be reused over and over again.

So if you have an SQL query that you write over and over again, save it as a stored procedure, and then just call it to execute it.

You can also pass parameters to a stored procedure, so that the stored procedure can act based on the parameter value(s) that is passed.

-- -- Create Database

CREATE DATABASE YourDatabaseName;

USE YourDatabaseName;

-- Create Table

CREATE TABLE Customers (

 CustomerID INT PRIMARY KEY,

 CustomerName NVARCHAR(255),

 ContactName NVARCHAR(255),

 Address NVARCHAR(255),

 City NVARCHAR(255),

 PostalCode NVARCHAR(20),

 Country NVARCHAR(255)

);

-- Insert Values

INSERT INTO Customers (CustomerID, CustomerName, ContactName, Address, City, PostalCode, Country)

VALUES

(1, 'Alfreds Futterkiste', 'Maria Anders', 'Obere Str. 57', 'Berlin', '12209', 'Germany'),
(2, 'Ana Trujillo Emparedados y helados', 'Ana Trujillo', 'Avda. de la Constitución 2222', 'México D.F.', '05021', 'Mexico'),
(3, 'Antonio Moreno Taquería', 'Antonio Moreno', 'Mataderos 2312', 'México D.F.', '05023', 'Mexico'),
(4, 'Around the Horn', 'Thomas Hardy', '120 Hanover Sq.', 'London', 'WA1 1DP', 'UK'),
(5, 'Berglunds snabbköp', 'Christina Berglund', 'Berguvsvägen 8', 'Luleå', 'S-958 22', 'Sweden');

-- Stored Procedure: Select All Customers

```
CREATE PROCEDURE SelectAllCustomers  
  
AS  
  
BEGIN  
  
    SELECT * FROM Customers;  
  
END;
```

-- Stored Procedure: Select Customers by City

```
CREATE PROCEDURE SelectCustomersByCity  
  
    @City NVARCHAR(255)  
  
AS  
  
BEGIN  
  
    SELECT * FROM Customers WHERE City = @City;  
  
END;
```

-- Stored Procedure: Select Customers by City and Postal Code

```
CREATE PROCEDURE SelectCustomersByCityAndPostalCode  
  
    @City NVARCHAR(255),
```

@PostalCode NVARCHAR(20)

AS

BEGIN

SELECT * FROM Customers WHERE City = @City AND PostalCode = @PostalCode;

END;

-- Execute the stored procedures

EXEC SelectAllCustomers;

EXEC SelectCustomersByCity @City = 'London';

EXEC SelectCustomersByCityAndPostalCode @City = 'London', @PostalCode = 'WA1 1DP';