

SQL ALTER TABLE Statement

The SQL ALTER TABLE command is used to modify the definition (structure) of a table by modifying the definition of its columns. The ALTER command is used to perform the following functions.

- 1) Add, drop, and modify table columns
- 2) Add drop constraints
- 3) Enable and Disable constraints

Syntax to add a column

ALTER TABLE table_name ADD column_name datatype;

For Example: To add a column "experience" to the employee table, the query would be like

ALTER TABLE employee ADD experience number(3);

Syntax to drop a column

ALTER TABLE table_name DROP column_name;

For Example: To drop the column "location" from the employee table, the query would be like

ALTER TABLE employee DROP location;

Syntax to modify a column

ALTER TABLE table_name MODIFY column_name datatype;

For Example: To modify the column salary in the employee table, the query would be like

ALTER TABLE employee MODIFY salary number(15,2);

SQL RENAME Command

The SQL RENAME command is used to change the name of the table or a database object.

If you change the object's name any reference to the old name will be affected. You have to manually change the old name to the new name in every reference.

Syntax to rename a table

RENAME old_table_name To new_table_name;

For Example: To change the name of the table employee to my_employee, the query would be like

RENAME employee TO my_employee;

SQL JOINS: Exercise-1 with Solution

Write a SQL statement to prepare a list with salesman name, customer name and their cities for the salesmen and customer who belongs to the same city.

Sample table: salesman

salesman_id	name	city	commission
5001	James Hoog	New York	0.15
5002	Nail Knite	Paris	0.13
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5003	Lauson Hen		0.12
5007	Paul Adam	Rome	0.13

Sample table: customer

customer_id	cust_name	city	grade	salesman_id
3002	Nick Rimando	New York	100	5001
3005	Graham Zusi	California	200	5002
3001	Brad Guzan	London		5005
3004	Fabian Johns	Paris	300	5006
3007	Brad Davis	New York	200	5001
3009	Geoff Camero	Berlin	100	5003
3008	Julian Green	London	300	5002
3003	Jozy Altidor	Moscow	200	5007

Sample Solution:

```
SELECT salesman.name AS "Salesman",
customer.cust_name, customer.city
FROM salesman, customer
WHERE salesman.city=customer.city;
```

Copy

Output of the Query:

Salesman	cust_name	city
James Hoog	Nick Rimando	New York
James Hoog	Brad Davis	New York
Pit Alex	Julian Green	London

Mc Lyon	Fabian Johnson	Paris
Nail Knite	Fabian Johnson	Paris
Pit Alex	Brad Guzan	London

Explanation:

Pictorial presentation:

```

SELECT salesman.name AS "Salesman",
customer.cust_name, customer.city
FROM salesman, customer
WHERE salesman.city=customer.city;

```

salesman

salesman_id	name	city	commission
5001	James Hoog	New York	0.15
5002	Nail Knite	Paris	0.13
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5003	Lauson Hense		0.12
5007	Paul Adam	Rome	0.13

customer

customer_id	cust_name	city	grade	salesman_id
3002	Nick Rimando	New York	100	5001
3005	Graham Zusi	California	200	5002
3004	Fabian Johnson	Paris	300	5006
3007	Brad Davis	New York	200	5001
3009	Geoff Cameron	Berlin	100	5003
3008	Julian Green	London	300	5002
3001	Brad Guzan	London		5005
3003	Jozy Altidore	Moscow	200	5007

salesman

name	city
James Hoog	New York

customer

city	cust_name
New York	Nick Rimando
New York	Brad Davis

name	city
Nail Knite	Paris
Mc Lyon	Paris

city	cust_name
Paris	Fabian Johnson

name	city
Pit Alex	London

city	cust_name
London	Julian Green
London	Brad Guzan

Salesman	cust_name	city
James Hoog	Brad Davis	New York
James Hoog	Nick Rimando	New York
Nail Knite	Fabian Johnson	Paris
Pit Alex	Brad Guzan	London
Pit Alex	Julian Green	London
Mc Lyon	Fabian Johnson	Paris

SQL JOINS: Exercise-4 with Solution

Write a SQL statement to find the list of customers who appointed a salesman for their jobs who gets a commission from the company is more than 12%.

Sample table: customer

customer_id	cust_name	city	grade	salesman_id
3002	Nick Rimando	New York	100	5001
3005	Graham Zusi	California	200	5002
3001	Brad Guzan	London		5005
3004	Fabian Johns	Paris	300	5006
3007	Brad Davis	New York	200	5001
3009	Geoff Camero	Berlin	100	5003
3008	Julian Green	London	300	5002
3003	Jozy Altidor	Moscow	200	5007

Sample table: salesman

salesman_id	name	city	commission
5001	James Hoog	New York	0.15
5002	Nail Knite	Paris	0.13
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5003	Lauson Hen		0.12
5007	Paul Adam	Rome	0.13

Sample Solution:

```
SELECT a.cust_name AS "Customer Name",
a.city, b.name AS "Salesman", b.commission
FROM customer a
INNER JOIN salesman b
ON a.salesman_id=b.salesman_id
WHERE b.commission>.12;
```

Copy

Output of the Query:

Customer Name	city	Salesman	commission
Nick Rimando	New York	James Hoog	0.15

Brad Davis	New York	James Hoog	0.15	
Graham Zusi	California	Nail Knite	0.13	
Julian Green	London	Nail Knite	0.13	
Fabian Johnson	Paris	Mc Lyon		0.14
Jozy Altidor	Moscow	Paul Adam	0.13	

Explanation:

```
SELECT a.cust_name AS "Customer Name",
a.city, b.name AS "Salesman", b.commission
FROM customer a
INNER JOIN salesman b
ON a.salesman_id=b.salesman_id
WHERE b.commission>.12;
```

Pictorial presentation:

```

SELECT a.cust_name AS "Customer Name",
a.city, b.name AS "Salesman", b.commission
FROM customer a
INNER JOIN salesman b
ON a.salesman_id=b.salesman_id
WHERE b.commission>.12;

```

customer

customer_id	cust_name	city	grade	salesman_id
3002	Nick Rimando	New York	100	5001
3005	Graham Zusi	California	200	5002
3004	Fabian Johnson	Paris	300	5006
3007	Brad Davis	New York	200	5001
3009	Geoff Cameron	Berlin	100	5003
3008	Julian Green	London	300	5002
3001	Brad Guzan	London		5005
3003	Jozy Altidore	Moscow	200	5007

salesman

salesman_id	name	city	commission
5001	James Hoog	New York	0.15
5002	Nail Knite	Paris	0.13
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5003	Lauson Hense		0.12
5007	Paul Adam	Rome	0.13

customer

cust_name	city	..	salesman_id
Nick Rimando	New York	..	5001
Brad Davis	New York	..	5001

salesman

salesman_id	name	commission
5001	James Hoog	0.15

cust_name	city	..	salesman_id
Graham Zusi	California	..	5002
Julian Green	London	..	5002

salesman_id	name	commission
5002	Nail Knite	0.13

cust_name	city	..	salesman_id
Fabian Johnson	Paris	..	5006

salesman_id	name	commission
5006	Mc Lyon	0.14

cust_name	city	..	salesman_id
Geoff Cameron	Berlin	..	5003

salesman_id	name	commission
5003	Lauson Hense	0.12

cust_name	city	..	salesman_id
Brad Guzan	London	..	5005

salesman_id	name	commission
5005	Pit Alex	0.11

cust_name	city	..	salesman_id
Jozy Altidore	Moscow	..	5007

salesman_id	name	commission
5007	Paul Adam	0.13

Customer Name	city	Salesman	commission
Nick Rimando	New York	James Hoog	0.15
Graham Zusi	California	Nail Knite	0.13
Fabian Johnson	Paris	Mc Lyon	0.14
Brad Davis	New York	James Hoog	0.15
Julian Green	London	Nail Knite	0.13

SQL JOINS: Exercise-5 with Solution

Write a SQL statement to find the list of customers who appointed a salesman for their jobs who does not live in the same city where their customer lives, and gets a commission is above 12%.

Sample table: customer

customer_id	cust_name	city	grade	salesman_id
3002	Nick Rimando	New York	100	5001
3005	Graham Zusi	California	200	5002
3001	Brad Guzan	London		5005
3004	Fabian Johns	Paris	300	5006
3007	Brad Davis	New York	200	5001
3009	Geoff Camero	Berlin	100	5003
3008	Julian Green	London	300	5002
3003	Jozy Altidor	Moscow	200	5007

Sample table: salesman

salesman_id	name	city	commission
5001	James Hoog	New York	0.15
5002	Nail Knite	Paris	0.13
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5003	Lauson Hen		0.12
5007	Paul Adam	Rome	0.13

Sample Solution:

```
SELECT a.cust_name AS "Customer Name",
a.city, b.name AS "Salesman", b.city,b.commission
FROM customer a
INNER JOIN salesman b
ON a.salesman_id=b.salesman_id
WHERE b.commission>.12
AND a.city<>b.city;
```


Copy

Output of the Query:

Customer Name	city	Salesman	city	commission
Graham Zusi	California	Nail Knite	Paris	0.13
Julian Green	London	Nail Knite	Paris	0.13
Jozy Altidor	Moscow	Paul Adam	Rome	0.13

Explanation:

```
SELECT a.cust_name AS "Customer Name", a.city,
b.name AS "Salesman", b.city,b.commission
FROM customer a
INNER JOIN salesman b
ON a.salesman_id=b.salesman_id
WHERE b.commission>.12
AND a.city<>b.city;
```

Pictorial presentation:

```

SELECT a.cust_name AS "Customer Name", a.city,
b.name AS "Salesman", b.city, b.commission
FROM customer a
INNER JOIN salesman b
ON a.salesman_id=b.salesman_id
WHERE b.commission>.12
AND a.city<>b.city;

```

customer

customer_id	cust_name	city	grade	salesman_id
3002	Nick Rimando	New York	100	5001
3005	Graham Zusi	California	200	5002
3004	Fabian Johnson	Paris	300	5006
3007	Brad Davis	New York	200	5001
3009	Geoff Cameron	Berlin	100	5003
3008	Julian Green	London	300	5002
3001	Brad Guzan	London		5005
3003	Jozy Altidore	Moscow	200	5007

salesman

salesman_id	name	city	commission
5001	James Hoog	New York	0.15
5002	Nail Knite	Paris	0.13
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5003	Lauson Hense		0.12
5007	Paul Adam	Rome	0.13

customer

cust_name	city	..	salesman_id
Nick Rimando	New York	..	5001
Brad Davis	New York	..	5001

salesman

salesman_id	name	city	commission
5001	James Hoog	New York	0.15

cities are same

cust_name	city	..	salesman_id
Graham Zusi	California	..	5002
Julian Green	London	..	5002

salesman_id	name	city	commission
5002	Nail Knite	Paris	0.13

cities are same

cust_name	city	..	salesman_id
Fabian Johnson	Paris	..	5006

salesman_id	name	city	commission
5006	Mc Lyon	Paris	0.14

cust_name	city	..	salesman_id
Geoff Cameron	Berlin	..	5003

salesman_id	name	city	commission
5003	Lauson Hense		0.12

cust_name	city	..	salesman_id
Brad Guzan	London	..	5005

salesman_id	name	city	commission
5005	Pit Alex	London	0.11

cust_name	city	..	salesman_id
Jozy Altidore	Moscow	..	5007

salesman_id	name	city	commission
5007	Paul Adam	Rome	0.13

Customer Name	city	Salesman	city	commission
Graham Zusi	California	Nail Knite	Paris	0.13
Julian Green	London	Nail Knite	Paris	0.13
Jozy Altidore	Moscow	Paul Adam	Rome	0.13

SQL JOINS: Exercise-2 with Solution

Write a SQL statement to make a list with order no, purchase amount, customer name and their cities for those orders which order amount between 500 and 2000.

Sample table: orders

ord_no	purch_amt	ord_date	customer_id	salesman_id
70001	150.5	2012-10-05	3005	5002
70009	270.65	2012-09-10	3001	5005
70002	65.26	2012-10-05	3002	5001
70004	110.5	2012-08-17	3009	5003
70007	948.5	2012-09-10	3005	5002
70005	2400.6	2012-07-27	3007	5001
70008	5760	2012-09-10	3002	5001
70010	1983.43	2012-10-10	3004	5006
70003	2480.4	2012-10-10	3009	5003
70012	250.45	2012-06-27	3008	5002
70011	75.29	2012-08-17	3003	5007
70013	3045.6	2012-04-25	3002	5001

Sample table: customer

customer_id	cust_name	city	grade	salesman_id
3002	Nick Rimando	New York	100	5001
3005	Graham Zusi	California	200	5002
3001	Brad Guzan	London		5005
3004	Fabian Johns	Paris	300	5006
3007	Brad Davis	New York	200	5001
3009	Geoff Camero	Berlin	100	5003
3008	Julian Green	London	300	5002
3003	Jozy Altidor	Moscow	200	5007

Sample Solution :

```
SELECT  a.ord_no,a.purch_amt,
b.cust_name,b.city
FROM orders a,customer b
WHERE a.customer_id=b.customer_id
AND a.purch_amt BETWEEN 500 AND 2000;
```

Copy

Output of the Query:

ord_no	purch_amt	cust_name	city
70007	948.50	Graham Zusi	California
70010	1983.43	Fabian Johnson	Paris

Explanation:

```

SELECT a.ord_no,a.purch_amt,
b.cust_name,b.city
FROM orders a,customer b
WHERE a.customer_id=b.customer_id
AND a.purch_amt BETWEEN 500 AND 2000;

```

Pictorial presentation:

```

SELECT a.ord_no,a.purch_amt,
b.cust_name,b.city
FROM orders a,customer b
WHERE a.customer_id=b.customer_id
AND a.purch_amt BETWEEN 500 AND 2000;

```

orders

ord_no	purch_amt	ord_date	customer_id	salesman_id
70001	150.50	2012-10-05	3005	5002
70009	270.65	2012-09-10	3001	5005
70002	65.26	2012-10-05	3002	5001
70004	110.50	2012-08-17	3009	5003
70007	948.50	2012-09-10	3005	5002
70005	2400.60	2012-07-27	3007	5001
70008	5760.00	2012-09-10	3002	5001
70010	1983.43	2012-10-10	3004	5006
70003	2480.40	2012-10-10	3009	5003
70012	250.45	2012-06-27	3008	5002
70011	75.29	2012-08-17	3003	5007
70013	3045.60	2012-04-25	3002	5001

customer

customer_id	cust_name	city	grade	salesman_id
3002	Nick Rimando	New York	100	5001
3005	Graham Zusi	California	200	5002
3004	Fabian Johnson	Paris	300	5006
3007	Brad Davis	New York	200	5001
3009	Geoff Cameron	Berlin	100	5003
3008	Julian Green	London	300	5002
3001	Brad Guzan	London		5005
3003	Jozy Altidore	Moscow	200	5007

orders

ord_no	purch_amt	ord_date	customer_id
70001	150.50	2012-10-05	3005
70009	270.65	2012-09-10	3001
70002	65.26	2012-10-05	3002
70004	110.50	2012-08-17	3009
70007	948.50	2012-09-10	3005
70005	2400.60	2012-07-27	3007
70008	5760.00	2012-09-10	3002
70010	1983.43	2012-10-10	3004

customer

customer_id	cust_name	city	grade	salesman_id
3002	Nick Rimando	New York	100	5001
3005	Graham Zusi	California	200	5002
3004	Fabian Johnson	Paris	300	5006
3007	Brad Davis	New York	200	5001
3009	Geoff Cameron	Berlin	100	5003
3008	Julian Green	London	300	5002
3001	Brad Guzan	London		5005
3003	Jozy Altidore	Moscow	200	5007

ord_no	purch_amt	cust_name	city
70007	948.50	Graham Zusi	California
70010	1983.43	Fabian Johnson	Paris

SQL JOINS: Exercise-3 with Solution

Write a SQL statement to know which salesman are working for which customer.

Sample table: customer

customer_id	cust_name	city	grade	salesman_id
3002	Nick Rimando	New York	100	5001
3005	Graham Zusi	California	200	5002
3001	Brad Guzan	London		5005
3004	Fabian Johns	Paris	300	5006
3007	Brad Davis	New York	200	5001
3009	Geoff Camero	Berlin	100	5003
3008	Julian Green	London	300	5002
3003	Jozy Altidor	Moscow	200	5007

Sample table: salesman

salesman_id	name	city	commission
5001	James Hoog	New York	0.15
5002	Nail Knite	Paris	0.13
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5003	Lauson Hen		0.12
5007	Paul Adam	Rome	0.13

Sample Solution:

```
SELECT a.cust_name AS "Customer Name",
a.city, b.name AS "Salesman", b.commission
FROM customer a
INNER JOIN salesman b
ON a.salesman_id=b.salesman_id;
```

Copy

Output of the Query:

Customer Name	city	Salesman	commission
Nick Rimando	New York	James Hoog	0.15
Brad Davis	New York	James Hoog	0.15
Graham Zusi	California	Nail Knite	0.13
Julian Green	London	Nail Knite	0.13
Fabian Johnson	Paris	Mc Lyon	0.14

Geoff Cameron	Berlin	Lauson Hen	0.12
Jozy Altidor	Moscow	Paul Adam	0.13
Brad Guzan	London	Pit Alex	0.11

Explanation:

```
SELECT a.cust_name AS "Customer Name",  
a.city,b.name AS "Salesman", b.commission  
FROM customer a  
INNER JOIN salesman b  
ON a.salesman_id=b.salesman_id;
```

Pictorial presentation :

```

SELECT a.cust_name AS "Customer Name",
a.city,b.name AS "Salesman", b.commission
FROM customer a
INNER JOIN salesman b
ON a.salesman_id=b.salesman_id;

```

customer

customer_id	cust_name	city	grade	salesman_id
3002	Nick Rimando	New York	100	5001
3005	Graham Zusi	California	200	5002
3004	Fabian Johnson	Paris	300	5006
3007	Brad Davis	New York	200	5001
3009	Geoff Cameron	Berlin	100	5003
3008	Julian Green	London	300	5002
3001	Brad Guzan	London		5005
3003	Jozy Altidore	Moscow	200	5007

salesman

salesman_id	name	city	commission
5001	James Hoog	New York	0.15
5002	Nail Knite	Paris	0.13
5005	Pit Alex	London	0.11
5006	Mc Lyon	Paris	0.14
5003	Lauson Hense		0.12
5007	Paul Adam	Rome	0.13

customer

cust_name	city	..	salesman_id
Nick Rimando	New York	..	5001
Brad Davis	New York	..	5001

salesman

salesman_id	name	commission
5001	James Hoog	0.15

cust_name	city	..	salesman_id
Graham Zusi	California	..	5002
Julian Green	London	..	5002

salesman_id	name	commission
5002	Nail Knite	0.13

cust_name	city	..	salesman_id
Fabian Johnson	Paris	..	5006

salesman_id	name	commission
5006	Mc Lyon	0.14

cust_name	city	..	salesman_id
Geoff Cameron	Berlin	..	5003

salesman_id	name	commission
5003	Lauson Hense	0.12

cust_name	city	..	salesman_id
Brad Guzan	London	..	5005

salesman_id	name	commission
5005	Pit Alex	0.11

cust_name	city	..	salesman_id
Jozy Altidore	Moscow	..	5007

salesman_id	name	commission
5007	Paul Adam	0.13

Customer Name	city	Salesman	commission
Nick Rimando	New York	James Hoog	0.15
Graham Zusi	California	Nail Knite	0.13
Fabian Johnson	Paris	Mc Lyon	0.14
Brad Davis	New York	James Hoog	0.15
Geoff Cameron	Berlin	Lauson Hense	0.12

