



## Untitled query, Feb 20, 2023

Created 8 days ago Schedule

Run all

Share

mysql

nada

```

31 INSERT into customer VALUES(3005,'grahm zusi','california',200,5002);
32 INSERT into customer VALUES(3008,'julian green','london',300,5002);
33 INSERT into customer VALUES(3004,'fabian johnson','paris',300,5006);
34 INSERT into customer VALUES(3009,'geoff cameron','berlin',100,5003);
35 INSERT into customer VALUES(3003,'jozy altidor','moscow',200,5007);
36
37 select salesmen.name as salesperson , customer.cust_name,customer.city
38 from salesmen
39 join customer
40 where salesmen.city=customer.city;
41
42
43
44 create table orders(
45     ord_no int primary key,
46     purch_amt decimal(8,3),
47     ord_date date,
48     customer_id int,
49     salesmen_id int,
50     foreign key(salesmen_id) references customer(salesmen_id) on delete cascade,
51     foreign key (customer_id) references customer(customer_id) on delete cascade
52 );
53
54 INSERT into orders VALUES(70001,150.5,'2012-10-05',3005,5002);
55 INSERT INTO orders VALUES(70009,270.65,"2012-09-10",3001,5005);
56 INSERT INTO orders VALUES(70002,65.26,"2012-10-05",3002,5001);
57 INSERT INTO orders VALUES(70004,110.5,"2012-09-10",3009,5003);
58 INSERT INTO orders VALUES(70007,948.5,"2012-09-10",3005,5002);
59 INSERT INTO orders VALUES(70005,2400.6,"2012-07-27",3007,5001);
60 INSERT INTO orders VALUES(70008,5760,"2012-09-10",3002,5001);

```

Clear all

|   |       |         |         |
|---|-------|---------|---------|
| 3 | 70007 | 948.500 | jozy al |
| 4 | 70007 | 948.500 | fabian  |
| 5 | 70007 | 948.500 | grahm   |
| 6 | 70007 | 948.500 | brade   |
| 7 | 70007 | 948.500 | julian  |
| 8 | 70007 | 948.500 | geoff   |

Untitled query, Feb 20, 2023

Success (5 rows) 0.1 s

7:10 PM

Explore SQL Data Chart Export

|   | on | cust_name      | city   |
|---|----|----------------|--------|
| 1 | g  | nick rimando   | new yc |
| 2 |    | fabian johnson | paris  |
| 3 |    | fabian johnson | paris  |
| 4 | g  | brade davis    | new yc |
| 5 |    | julian green   | london |

query, Feb 20, 2023

Created 8 days ago

Schedule



Run all

mys

```
45 ord_no int primary key,  
46 purch_amt decimal(8,3),  
47 ord_date date,  
48 customer_id int,  
49 salesmen_id int,  
50 foreign key(salesmen_id) references customer(salesmen_id) on delete cascade,  
51 foreign key (customer_id) references customer(customer_id) on delete cascade  
52 );  
53  
54 INSERT into orders VALUES(70001,150.5,'2012-10-05',3005,5002);  
55 INSERT INTO orders VALUES(70009,270.65,"2012-09-10",3001,5005);  
56 INSERT INTO orders VALUES(70002,65.26,"2012-10-05",3002,5001);  
57 INSERT INTO orders VALUES(70004,110.5,"2012-09-10",3009,5003);  
58 INSERT INTO orders VALUES(70007,948.5,"2012-09-10",3005,5002);  
59 INSERT INTO orders VALUES(70005,2400.6,"2012-07-27",3007,5001);  
60 INSERT INTO orders VALUES(70008,5760,"2012-09-10",3002,5001);  
61  
62  
63 SELECT ord_no , purch_amt , cust_name , city  
64 FROM orders  
65 join customer  
66 WHERE purch_amt BETWEEN 500 AND 2000;
```

Clear all

|   |       |         |
|---|-------|---------|
| 6 | 70007 | 948.500 |
| 7 | 70007 | 948.500 |
| 8 | 70007 | 948.500 |

Untitled query, Feb 20, 2023

Success (5 rows) 0.1 s

Explore SQL Data Chart

|   | on | cust_name      |
|---|----|----------------|
| 1 | g  | nick rimando   |
| 2 |    | fabian johnson |
| 3 |    | fabian johnson |
| 4 | g  | brade davis    |
| 5 |    | julian green   |

Untitled query, Feb 20, 2023

Success (8 rows) 0.1 s

OPPO Reno4

2023/02/28 21:50

شغل مذك

\* inner join \*

في حالة وجود Two Tables  
يقوم inner join بأخذ مشترك بينهم

outer join .. عبارة عن left, Right, Full

left .. يقوم بأخذ كل data موجود في

left وليس موجودة في Right

Right .. يقوم بأخذ كل data الموجودة في Right

وهي موجودة في left

Full .. يقوم بتجميع كل البيانات الموجودة في

الجدولين سواء كان فيه مشترك أو لا



مس الترتیب

Array تریج لایه اول و

اولی و تریج اول

تریج اول و تریج اول و تریج اول

تریج اول و تریج اول و تریج اول

تریج اول و تریج اول و تریج اول

تریج اول و تریج اول و تریج اول

تریج اول و تریج اول و تریج اول

تریج اول و تریج اول و تریج اول

Merg sort.

input → Algorithms → output

المدخلات

index, index

نوع الخوارزمية ووقت التنفيذ

الوقت الذي تستغرقه الخوارزمية

Linear Search → الوقت الذي تستغرقه الخوارزمية

$O(n)$

الوقت الذي تستغرقه الخوارزمية

$O(n^2)$

الوقت الذي تستغرقه الخوارزمية

الوقت الذي تستغرقه الخوارزمية

Array for loop

الوقت الذي تستغرقه الخوارزمية

for i from 0 to n-1

if number behind door[i]

Return true

Return false

Binary Search

الوقت الذي تستغرقه الخوارزمية

$O(\log n)$

time solve

size of problem

Binary search:

if no doors

Return false

if number behind doors [middle]

Return true

Else if number < doors [middle]

Search doors [0] through [middle-1]

Else if number > doors [middle]

Search doors [middle+1] through

doors [n-1]

Binary search will return Array [n]

Selection sort:

نبدأ من أول المصفوفة ونبحث عن أصغر عنصر ونضعه في مكانه الأول

swap index 0 مع index 1

نكرر العملية

n + n-1 + n-2 + n-3 + ...

$O(n^2)$

الوقت هو  $O(n^2)$

$n^2$

نبحث ونقارن

loop

loop



## Bubble Sort:

if number  $[i]$  and numbers  $[i+1]$  out of order swap them

$O(n^2)$

$O(n)$

Repeat  $n-1$  times

For  $i$  from  $0$  to  $n-2$

if numbers  $[i]$  and numbers  $[i+1]$  out of order

swap them

if no swaps

Quit

Recursion: void fun (int x)

{ fun(x);

}