

272 -ASSIGNMENT 2

Information in Cloud (Mini Projects)

KEVIN NGUYEN	008604206
PRANJAL SHARMA	013831688
RACHIT SAXENA	012469626
VINIT DHOLAKIA	013821379

A.) SQLite

SQLite provides a relational database management system. The lite in SQLite means light weight in terms of setup. It can be installed as a plugin in Mozilla Firefox.

Purchase Order Management System

Purchase Order management schema consists of three tables:

1. Supplier Table

create table supplier (supplierid INTEGER PRIMARY KEY, suppliername VARCHAR(30) NOT NULL).

```
create table supplier(supplierid INTEGER PRIMARY KEY,suppliername VARCHAR(30) NOT NULL)
no output
```

Sample Data:

insert into supplier values(1,'Target');

insert into supplier values(3,'Walmart');

2. Products Table

create table products (product_id INTEGER PRIMARY KEY, supplierid INTEGER,product_price VARCHAR(50), product_description VARCHAR(200)).

```
create table products(product_id INTEGER PRIMARY KEY,supplierid INTEGER,product_price VARCHAR(50),product_description VARCHAR(200))
no output
```

Sample Data:

insert into products values (30,3,'50\$','Market Pantry Cookies');

insert into products values (56,23,'60\$','Up and Up Skin Treatment');

insert into products values (23,1,'50\$','Room Essentials Crockery');

3. Customer Table

create table customer(customer_id INTEGER PRIMARY KEY , product_id INTEGER ,
customer_name varchar(50)) .

```
create table customer(customer_id INTEGER PRIMARY KEY,product_id INTEGER ,customer_name varchar(50))  
no output
```

Sample Data:

insert into customer values (20,30,'Kevin Nguyen');

insert into customer values (23,32,'Pranjal Sharma');

insert into customer values (32,23,'Rachit Saxena');

QUERIES PERFORMED ON SAMPLE DATA

INNER JOIN

select supplierid,product_price,product_description,customer_id,customer_name

from products inner join customer where products.product_id=customer.product_id.

```
select supplierid,product_price,product_description,customer_id,customer_name from products inner join customer on products.product_id=customer.product_id
```

Export	supplierid	product_price	product_description	customer_id	customer_name
1	1	50\$	Room Essentials Crockery	32	Rachit Saxena

Enter with id as SQLite command

RIGHT JOIN

select supplierid,product_price,product_description,customer_id,customer_name from
products right join customer on products.product_id=customer.product_id.

```
select supplierid,product_price,product_description,customer_id,customer_name from products right join customer on products.product_id=customer.product_id
```

RIGHT and FULL OUTER JOINS are not currently supported

LEFT JOIN

select supplierid,product_price,product_description,customer_id,customer_name from
products left join customer on products.product_id=customer.product_id

```
select supplierid,product_price,product_description,customer_id,customer_name from products left join customer on products.product_id=customer.product_id
```

Export	supplierid	product_price	product_description	customer_id	customer_name
1	1	50\$	Room Essentials Crockery	32	Rachit Saxena
2	3	50\$	Market Pantry Cookies	20	Kevin Nguyen
3	23	60\$	Up and Up Skin Treatment	NULL	NULL

Enter math.js or SQLite commands

UPDATE

UPDATE supplier

SET suppliername="Spar"

WHERE supplierid=1

ORDER supplierid

```
UPDATE supplier
SET suppliername="Spar"

WHERE
    supplierid=1
```

no output

```
select * from supplier;
```

Export	supplierid	suppliername
1	1	Spar
2	3	Walmart

Enter math.js or SQLite commands

ALTER TABLE

ALTER TABLE supplier

RENAME TO inventory;

```
ALTER TABLE supplier  
RENAME TO inventory;
```

no output

```
select * from inventory;
```

Export	supplierid	suppliername
1	1	Spar
2	3	Walmart

```
Enter math.js or SQLite commands
```

CROSS JOIN

select * from customer cross join products

```
select * from customer cross join products
```

Export	customer_id	product_id	customer_name	product_id	supplierid	product_price	product_description
1	20	30	Kevin Nguyen	23	1	50\$	Room Essentials Crockery
2	20	30	Kevin Nguyen	30	3	50\$	Market Pantry Cookies
3	20	30	Kevin Nguyen	56	23	60\$	Up and Up Skin Treatment
4	23	32	Pranjal Sharma	23	1	50\$	Room Essentials Crockery
5	23	32	Pranjal Sharma	30	3	50\$	Market Pantry Cookies
6	23	32	Pranjal Sharma	56	23	60\$	Up and Up Skin Treatment
7	32	23	Rachit Saxena	23	1	50\$	Room Essentials Crockery
8	32	23	Rachit Saxena	30	3	50\$	Market Pantry Cookies
9	32	23	Rachit Saxena	56	23	60\$	Up and Up Skin Treatment

BETWEEN

SELECT supplierid, product_id, product_price FROM products WHERE supplierid BETWEEN 1 and 10 ORDER BY supplierid

```
SELECT supplierid, product_id, product_price FROM products WHERE supplierid BETWEEN 1 and 10 ORDER BY supplierid
```

Export	supplierid	product_id	product_price
1	1	23	50\$
2	3	30	50\$

```
Enter math.js or SQLite commands
```

DROP

drop table inventory

```
drop table inventory
```

no output

```
select * from inventory
```

no such table: inventory

AGGREGATE FUNCTIONS

AVG

SELECT avg(product_price) FROM products

MAX

SELECT max(product_price) FROM products

MIN

SELECT min(product_price) FROM products

```
SELECT avg(product_price) FROM products
```

	Export	avg(product_price)
--	--------	--------------------

1		53.33333333333336
---	--	-------------------

```
SELECT max(product_price) FROM products
```

	Export	max(product_price)
--	--------	--------------------

1		60\$
---	--	------

```
SELECT min(product_price) FROM products
```

	Export	min(product_price)
--	--------	--------------------

1		50\$
---	--	------