

Order of Items

- SQL Insert**
- SQL Delete**
- SQL Update**
- SQL Queries**
- Website Pseudocode example**

INSERT (create entry)

Everyone Login Page

insert a user

insert into USER values (uname, nam, birth, ph, addr, pass, mail);

Managers Only Profile Page

insert a manager

insert into MANAGER values (uname, mID);

insert an employee

insert into EMPLOYEE values (uname, eID);

insert a customer

insert into CUSTOMER values (uname);

Managers Only Item Management Page

insert an item

insert into ITEMS values (pNum, bar, nam, desc, sale, purchase);

insert a location

insert into LOCATION values (sNum, nam, addr, ph, hour, dept);

insert a store

insert into STORE values (sNum);

insert a warehouse

insert into WAREHOUSE values (sNum);

insert an item into a location (located)

insert into LOCATED values (pNum, bar, sNum, quant);

Manager Only Purchase Order Page

insert a purchase_order

insert into PURCHASE_ORDER (orderID, oDate, createdBy, id) values (oID, date, created, mID);

insert a cart into a purchase_order (process)

insert into PROCESS values (cID, oID);

Everyone Cart Page

insert a cart

insert into CART (cartID, createdBy) values (cID, created);

insert an item into a cart (contains)

insert into CONTAINS values (cID, pNum, bar, quant);

insert an item into a cart (contains) but limited by available quantity

select if((quant <= (select sum(quantity) from LOCATED Lo where Lo.productNum=pNum and Lo.barcode=bar)), (insert into CONTAINS(cID, pNum, bar, quant)), ('invalid quantity selected'));

Insert an item into a cart Alternative format (doesnt quite work yet)

DELIMITER //

BEGIN

DECLARE sumItemQty = (sum(quantity) from LOCATED Lo where o.productNum=1234567890 and Lo.barcode=12345678)

if sumItemQty <= 20 then

insert into CONTAINS(1234567, 1234567890, 12345678, 500);

else

select 'not enough stock';

end if;

END //

DELIMITER ;

DELETE

Managers Only Profile Page

delete a user

delete from USER where username=uname;

delete an employee

delete from EMPLOYEE where id=eID;

delete a manager

delete from MANAGER where id=mID;

delete a customer

delete from CUSTOMER where username=username;

Managers Only Item Management Page

delete an item from the system

delete from ITEMS where productNum=pNum and barcode=bar;

delete a location

delete from LOCATION where storeNum=sNum;

delete a store

delete from STORE where storeNum=sNum;

delete a warehouse

delete from WAREHOUSE where storeNum=sNum;

remove an item from a location (located)

delete from LOCATED where productNum=pNum and barcode=\$bar and storeNum=sNum;

delete a cart

delete from CART where cartID=cID;

Managers Only Purchase Order Page

remove a cart from an order (process)

delete from PROCESS where cartID=cID and orderID=oID;

delete a purchase order

delete from PURCHASE_ORDER where orderID=oID;

Everyone on Cart Page

remove item from cart (contains) limited by username from cookie

delete from CONTAINS, CART C where cartID=cID and C.createdBy=username;;

delete a cart limited by username from cookie

delete from CART where cartID=cID and C.createdBy=username;

UPDATE

Users profile page

update a name number for a given username;

update USER set name='name' where username='name';

update a phone number for a given username;

update USER set phone='###-###-####' where username='name';

update a password for a given username

update USER set password=newPassword where username='name';

update a email for a given username

update USER set email='newEmail' where username='name';

update an address for a given username

update USER set address='address' where username='name';

update a birthdate for a given username

update USER set dob='yyyy-mm-dd' where username='name';

Manager Only updates order page

update a purchase_order date for a given orderID

update PURCHASE_ORDER set oDate='yyyy-mm-dd' where orderID=number;

update a cartID for a purchase order

update PURCHASE_ORDER set cartID=number where cartID=number

Managers Only item management page

update the name of an ITEM

update ITEMS set name='name' where productNum=productN and barcode = barc;

update the description of an ITEM

update ITEMS set description=descrip where productNum=num and barcode =num2;

update the purchasePrice of an ITEM

update ITEMS set purchasePrice=price where productNum=num and barcode=num2;

update the salesPrice of an ITEM

update ITEMS set salesPrice=price where productNum=num and barcode=num2;

Managers and Employees on item management page

update the quantity of an ITEM

update ITEMS I, LOCATED L set quantity=newQuantity where I.productNum=L.productNum and I.barcode=L.barcode and L.productNum =prodNum and L.barcode=bar' and L.storeNum=snum;

Everyone on cart page

update quantity of a given item in a cart(ID)

update CONTAINS set quantity=quan where cartID=ID and productNum=prodNum and barcode=barc;

update the totalPrice of a cart

update CART set totalPrice =(select sum(salesPrice*quantity) totalPrice from ITEMS I, CONTAINS C where I.productNum=C.productNum and I.barcode=C.barcode and C.cartID='1234567') where CART.cartID='1234567';

update the totalPrice of a purchase_order

update PURCHASE_ORDER O set totalPrice=(select sum(C.totalPrice) from CART C, PROCESS P where P.cartID=C.cartID and P.orderID='123456') where O.orderID='123456';

QUERIES

Employees & Managers info page

find name, phone, email of all current managers

select name, phone, email from USER U
where exists
(select * from MANAGER M where M.username=U.username);

find name, phone, email of all current employees

select name, phone, email from USER U
where exists
(select * from EMPLOYEE E where E.username=U.username);

for a cart output the name, phone, email of customer that has an item in the cart (managers only)

select name, phone, email from USER U
where exists
(select * from CART C where C.createdby=U.username and cartID=number);

find contact information for current customers

select * from USER U where not exists

```
(select * from EMPLOYEE E where E.username = U.username)
and not exists
(select * from MANAGER M where M.username = U.username);
```

find contact information for current customers

```
select * from USER U where exists
(select * from EMPLOYEE E where E.username = U.username)
or exists
(select * from MANAGER M where M.username = U.username);
```

Managers on cart page

find an item by name or description (searching item in name and description)

```
select * from ITEMS
where name LIKE '%searchTerm%' or description LIKE '%searchTerm%';
```

Managers on purchase order page

find unprocessed carts

```
select * from CART where not exists ( select * from PROCESS where cartID=CART.cartID);
```

find processed carts

```
select * from CART where exists ( select * from PROCESS where cartID=CART.cartID);
```

a list of the items in all the purchase_orders

```
select name, sum(quantity) from ITEMS I, PROCESS P, CONTAINS C where
I.productNum=C.productNum and P.cartID =C.cartID and I.barcode=C.barcode group by
I.productNum, I.barcode;
```

find all carts

```
select cartID from CART;
```

list all items in a cart

```
select * from CONTAINS where cartID=$cart;
```

location and quantity of an item at a location

```
select address, sum(quantity) from LOCATION L, LOCATED Lo where Lo.storeNum=
L.storeNum and Lo.productNum =productNum and Lo.barcode = barcode group by L.storeNum;
```

Everyone on cart page

find an item by name (for customer, as purchase price is limited to managers)

```
select productNum, barcode, name, description, salesPrice from ITEMS
where name LIKE '%searchTerm%';
```

find an item by description or name (searching item in description and name)

```
select productNum, barcode, name, description,salesPrice from ITEMS
where description LIKE '%searchTerm%' or name LIKE '%searchTerm%';
```

find all Location (stores only)

```
select address from LOCATION L where exists
(select * from STORE S where S.storeNum = L.storeNum);
```

a list of the items in all the purchase_orders (limited to user)

```
select name, sum(quantity), salesPrice from ITEMS I, CONTAINS C, CART ca where
I.productNum=C.productNum and I.barcode=C.barcode and C.cartID=cart and
C.createdby=username group by I.productNum, I.barcode;
```

find all carts made by the user (limited to user only, they cannot see carts made by others) username comes from the cookie

```
select cartID from CART C where C.createdBy=username;
```

list all items in a cart where username comes from the cookie

```
select * from CONTAINS where cartID=cart and C.createdBy=username;
```

location and quantity of an item in a store

```
select address, sum(quantity) from LOCATION L, STORE S, LOCATED Lo where S.storeNum
= L.storeNum and S.storeNum= Lo.storeNum and Lo.productNum =productNum and
Lo.barcode = barcode group by L.storeNum;
```

find the sum of the quantity of items (total price) in a specific cart(with cartID=\$cart)

```
select sum(salesPrice*quantity) totalPrice from ITEMS I, CONTAINS C where
I.productNum=C.productNum and I.barcode=C.barcode and C.cartID=$cart;
```

find the sum of totalPrices of all carts within an order (find purchase_order totalPrice)

```
select sum(C.totalPrice) from CART C, PROCESS P where P.cartID=C.cartID and
P.orderID='123456');
```

Example Website pseudocode

Insert page pseudocode

insert an item

//insert_item.html

```
<!DOCTYPE>
<html>
<body>
<form action="insert_item.php" method="post">
Product No.:<input type="text" name="pNum"><br>
Barcode:<input type="text" name="bar"><br>
Name:<input type="text" name="name"><br>
Description:<input type="text" name="desc"><br>
Sales Price:<input type="text" name="sale"><br>
Purchase Price:<input type="text" name="purchase"><br>
<input type="submit">
</form>
</body>
</html>
```

```
//insert_item.php
<?php
$sql = "insert into ITEMS (productNum, barcode, name, description, salesPrice, purchasePrice)
values ('$_POST[pNum]', '$_POST[bar]', '$_POST[name]', '$_POST[desc]', '$_POST[sale]',
'$_POST[purchase]')";
?>
```

Delete page pseudocode

delete an item

```
//delete_item.html
<!DOCTYPE>
<html>
<body>
<form action="delete_item.php" method="post">
Product No.:<input type="text" name="pNum"><br>
Barcode:<input type="text" name="bar"><br>
<input type="submit">
</form>
</body>
</html>
```

```
//delete_item.php
<?php
$sql = "delete from ITEMS where (productNum='$_POST[pNum]' and barcode='$_POST[bar]')";
?>
```


Update page psuedo-code

update an item

```
//update_item.html
<!DOCTYPE>
<html>
<body>
<form action="update_item.php" method="post">
Product No.:<input type="text" name="pNum"><br>
Barcode:<input type="text" name="bar"><br><br>
New Info:<br>
Product No.:<input type="text" name="new_pNum"><br>
Barcode:<input type="text" name="new_bar"><br>
Name:<input type="text" name="name"><br>
Description:<input type="text" name="desc"><br>
Sales Price:<input type="text" name="sale"><br>
Purchase Price:<input type="text" name="purchase"><br>
<input type="submit">
</form>
</body>
</html>
```

```
//update_item.php
<?php
$sql = "update ITEMS set (productNum='$_POST[new_pNum]', barcode='$_POST[new_bar]',
name='$_POST[name]', description='$_POST[desc]', salesPrice='$_POST[sale]',
purchasePrice='$_POST[purchase]') where ('$_POST[pNum]' and '$_POST[bar]')";
?>
```

Query page psuedo-code

lookup an item

```
//query_item.html
<!DOCTYPE>
<html>
<body>
<form action="query_item.php" method="post">
Product No.:<input type="text" name="new_pNum"><br>
Barcode:<input type="text" name="new_bar"><br>
Name:<input type="text" name="name"><br>
Description:<input type="text" name="desc"><br>
Sales Price:<input type="text" name="sale"><br>
```

```
Purchase Price:<input type="text" name="purchase"><br>
<input type="submit">
</form>
</body>
</html>
```

```
//query_item.php
```

```
<?php
```

```
$sql = "select * from ITEMS where (productNum='$_POST[new_pNum]' or
barcode='$_POST[new_bar]' or name='$_POST[name]' or description='$_POST[desc]' or
salesPrice='$_POST[sale]' or purchasePrice='$_POST[purchase]')";
?>
```