



BY SHAUN GODFREY AND VITALIY SHYDLONOK

<https://isc329toolshop.000webhostapp.com/index.php>

ABOUT OUR DATABASE:

- Our project is a web-based Tool shop that allows its users to create an account and to search for or buy tools from our online store.
- Our Users will be able to search through our database for specific items based on their category, price, group, and terms in the name or description of the item they are looking for.
- Our Users will have a shopping cart with the items that they have chosen to purchase and through their account will also be able to view any order details.
- Lastly, our database will keep information from our user's account and details on the price and quantity of any item in our store.

ACCOMPLISHMENTS:

- The database incorporates the customer's name, email, card info, password, billing address, purchase history, and a current list of items in their shopping cart.
- The information saved for each item in our database is the name, description, price, quantity, review, and rating.
- The users will also be able to view the details of an item, add and remove items to their shopping cart, purchase items, and rate any items that they have purchased.
- Lastly, all orders will have details of the users, their purchase totals, the date of the order, and a record of where it was delivered too.

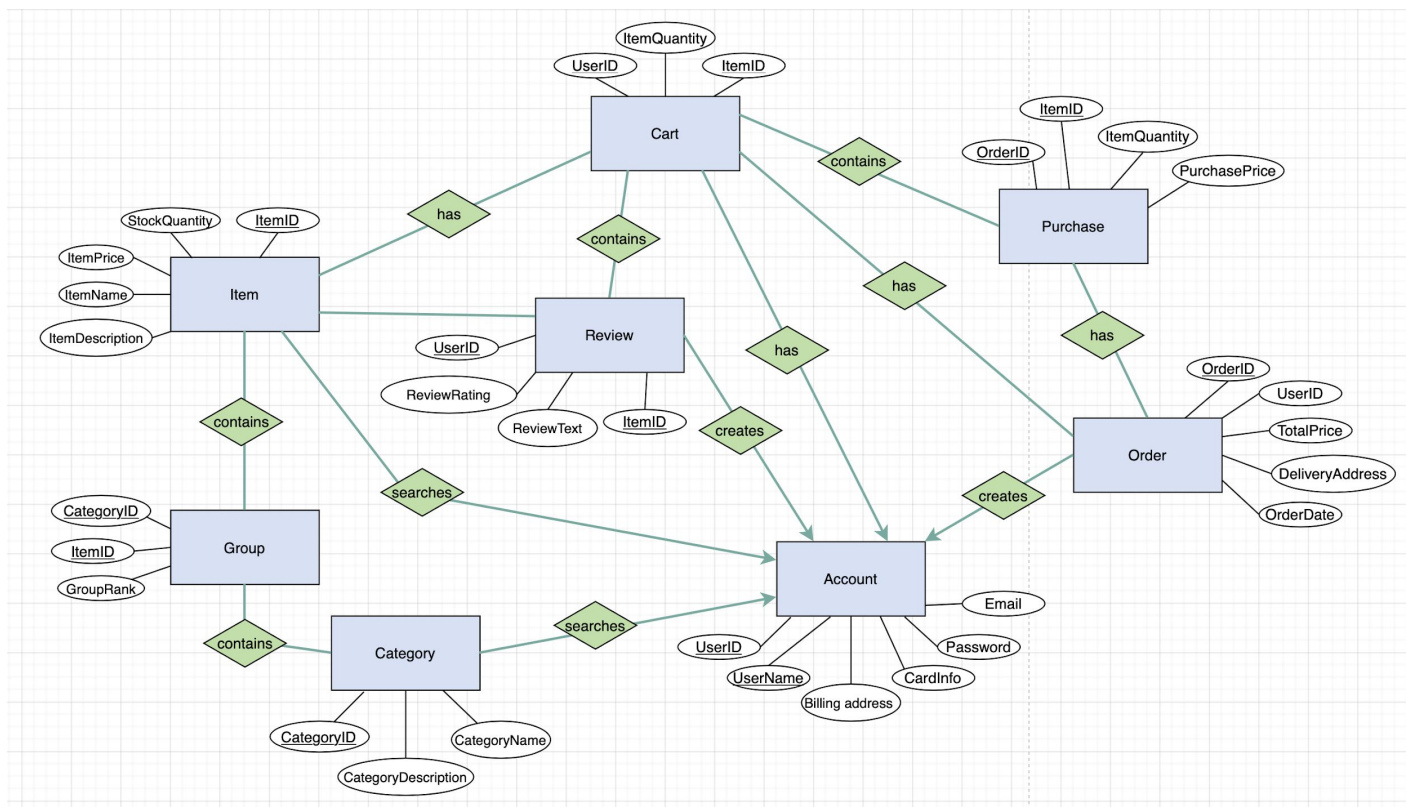
DESIGN SPECIFICATIONS

- When it came to the design of our database we started with the ER diagram, which helped to lay out the basic entity sets and their attributes that we needed.
- This helped also to define their relationships with each other and which attributes within this design would be our primary keys.
- We then designed the relational schema and realized that we needed more attributes for each of our entity sets, and some more entities as well. We thought we had everything figured out until we did the normalization process.

DESIGN SPECIFICATIONS

- During this process, we realized that we only had UserID as the main identifier for a User and that two accounts could have the same username. So, we fixed this by adding a UserName as a primary key to the ACCOUNT.
- We also added attributes Email and CardInfo to the ACCOUNT entity and added in the entity GROUP with attributes CategoryID, ItemID, and GroupRank, as each ITEM can have multiple values for CategoryID. We also removed the attributes CartID, PurchaseID, CategoryID, ReviewID, and ItemRating for our other entities as they were not needed and could be reached using other attributes.

ER DIAGRAM



RELATIONAL SCHEMA

UserID	UserName	Email	Password	CardInfo	BillingAddress
---------------	-----------------	-------	----------	----------	----------------

ACCOUNT

Bold: Primary Key
Underline: Foreign Key

<u>ItemID</u>	<u>UserID</u>	ItemQuantity
---------------	---------------	--------------

CART

OrderID	<u>ItemID</u>	ItemQuantity	PurchasePrice
----------------	---------------	--------------	---------------

PURCHASE

OrderID	<u>UserID</u>	TotalPrice	DeliveryAddress	OrderDate
----------------	---------------	------------	-----------------	-----------

ORDER

ItemID	ItemName	ItemDescription	ItemPrice	StockQuantity
---------------	----------	-----------------	-----------	---------------

ITEM

CategoryID	CategoryName	CategoryDescription
-------------------	--------------	---------------------

CATEGORY

<u>CategoryID</u>	<u>ItemID</u>	GroupRank
-------------------	---------------	-----------

GROUP

<u>UserID</u>	<u>ItemID</u>	ReviewRating	ReviewText
---------------	---------------	--------------	------------

REVIEW

CREATING THE DATABASE IN SQL 1/4

```
create table account
(
    UserID int auto_increment,
    UserName char(255) not null,
    Email char(255) not null,
    Password char(255) not null,
    CardInfo text null,
    BillingAddress text null,
    primary key (UserID, UserName)
);

create table item
(
    ItemID int auto_increment
        primary key,
    ItemName char(255) default 'Item Name' not null,
    ItemDescription text not null,
    ItemPrice double(16,2) not null,
    StockQuantity int default 0 not null
);
```


CREATING THE DATABASE IN SQL 2/4

```
create table cart
(
    ItemID int not null,
    UserID int not null,
    ItemQuantity int default 1 not null,
    primary key (ItemID, UserID),
    foreign key (UserID) references account (UserID),
    foreign key (ItemID) references item (ItemID)
);

create table category
(
    CategoryID int auto_increment
        primary key,
    CategoryName char(255) default 'Default Category' not null,
    CategoryDescription text not null
);
```

CREATING THE DATABASE IN SQL 3/4

```
create table group_table
(
    CategoryID int not null,
    ItemID int not null,
    GroupRank int not null,
    primary key (CategoryID, ItemID),
    foreign key (CategoryID) references category (CategoryID),
    foreign key (ItemID) references item (ItemID)
);

create table order_table
(
    OrderID int auto_increment
        primary key,
    UserID int not null,
    TotalPrice double(16,2) not null,
    DeliveryAddress text not null,
    OrderDate date not null,
    foreign key (UserID) references account (UserID)
);
```

CREATING THE DATABASE IN SQL 4/4

```
create table purchase
(
    OrderID int not null,
    ItemID int not null,
    ItemQuantity int default 1 not null,
    PurchasePrice double(16,2) not null,
    primary key (OrderID, ItemID),
    foreign key (ItemID) references item (ItemID),
    foreign key (OrderID) references order_table (OrderID)
);
```

```
create table review
(
    UserID int not null,
    ItemID int not null,
    ReviewRating int default 5 not null,
    ReviewText text not null,
    primary key (UserID, ItemID),
    foreign key (UserID) references account (UserID),
    foreign key (ItemID) references item (ItemID)
);
```

SQL QUERIES:

```
DELETE FROM cart WHERE ItemID = ? AND UserID = (SELECT UserID FROM account  
WHERE UserName = ?)
```

```
DELETE FROM cart WHERE UserID = (SELECT UserID FROM account WHERE UserName = ?)
```

```
DELETE FROM category WHERE CategoryID = ?
```

```
DELETE FROM item WHERE ItemID = ?
```

```
DELETE FROM review WHERE ItemID = ? AND UserID = ?
```

```
INSERT INTO account (UserName,Password,Email,CardInfo,BillingAddress) VALUES  
(?,?,?,?/?)
```

```
INSERT INTO cart (ItemID,UserID,ItemQuantity) VALUES (?(SELECT UserID FROM  
account WHERE UserName = ?),1)
```

```
INSERT INTO category (CategoryName,CategoryDescription) VALUES (?,?)
```

```
INSERT INTO group_table (CategoryID,ItemID,GroupRank) VALUES (?( SELECT  
LAST_INSERT_ID() ),1)
```

```
INSERT INTO item (ItemName,ItemDescription,ItemPrice,StockQuantity) VALUES  
(?,?,?,?)
```

```
INSERT INTO order_table (UserID,TotalPrice,DeliveryAddress,OrderDate) VALUES  
(?,?,?,?)
```

```
INSERT INTO purchase (OrderID,ItemID,ItemQuantity,PurchasePrice) VALUES  
(?,?,?,?)
```

```
INSERT INTO review (UserID,ItemID,ReviewRating,ReviewText) VALUES ((SELECT  
UserID FROM account WHERE UserName = ?),?,?,?)
```

```
SELECT * FROM account WHERE Username = ?
```

```
SELECT * FROM account WHERE Username = ? and Password = ?
```

```
SELECT * FROM cart WHERE ItemID = ? AND UserID = (SELECT UserID FROM account  
WHERE UserName = ?)
```

SQL QUERIES:

```
SELECT * FROM cart WHERE UserID = (SELECT UserID FROM account WHERE UserName = ?)
```

```
SELECT * FROM category
```

```
SELECT * FROM item WHERE ItemID = (SELECT ItemID FROM group_table WHERE CategoryID = ?)
```

```
SELECT * FROM item WHERE ItemID = ?
```

```
SELECT * FROM order_table WHERE UserID = (SELECT UserID FROM account WHERE UserName = ?)
```

```
SELECT * FROM purchase WHERE OrderID = ?
```

```
SELECT * FROM review WHERE ItemID = ? AND UserID = (SELECT UserID FROM account WHERE UserName = ?)
```

```
SELECT account.UserID as UserID, UserName, ReviewRating, ReviewText FROM review,account WHERE ItemID = ? AND review.UserID = account.UserID
```

```
SELECT CategoryName FROM category WHERE CategoryID = ?
```

```
SELECT ItemName FROM item WHERE ItemID = ?
```

```
SELECT ItemPrice FROM item WHERE ItemID = ?
```

```
SELECT SUM(ReviewRating) as ReviewTotal FROM review WHERE ItemID = ? AND UserID = (SELECT UserID FROM account WHERE UserName = ?)
```

```
SELECT UserID,SUM(ItemTotal) as TotalPrice FROM (SELECT UserID,ItemPrice*ItemQuantity as ItemTotal FROM (SELECT account.UserID as UserID,ItemPrice,ItemQuantity FROM item,card,account WHERE item.ItemID = card.ItemID and card.UserID = account.UserID and account.UserName = ?) AS PriceQuantity) AS TotalTable
```

```
UPDATE cart SET ItemQuantity = ItemQuantity + 1 WHERE ItemID = ? AND UserID = (SELECT UserID FROM account WHERE UserName = ?)
```

```
UPDATE cart SET ItemQuantity = ItemQuantity - 1 WHERE ItemID = ? AND UserID = (SELECT UserID FROM account WHERE UserName = ?)
```

```
UPDATE item SET StockQuantity = StockQuantity - ? WHERE ItemID = ?
```

```
UPDATE review SET ReviewRating = ?, ReviewText = ? WHERE ItemID = ? AND UserID = (SELECT UserID FROM account WHERE UserName = ?)
```

SCREENSHOTS OF OUR DATABASE: REGISTRATION

Tools Shop

Register

User Name:

Password:

Email:

Payment Card:

Billing Address:

SCREENSHOTS OF OUR DATABASE: LOGIN

Tools Shop

Login

User Name:

Password:

SCREENSHOTS OF OUR DATABASE: ACCOUNT

Account:

User: Admin

Email: test@a.com

Billing Address: test

[- Go To Cart -](#)

[- Previous Orders -](#)

[- Log Out -](#)

SCREENSHOTS OF OUR DATABASE: CATEGORIES LIST

Categories:

Woodworking: Tools for working with wood. | (Admin: Delete Category)

Welding: Tools used for welding | (Admin: Delete Category)

Power-tools: Tools that require power to operate. | (Admin: Delete Category)

Soldering: Tools used for soldering | (Admin: Delete Category)

(Admin) Add Category:

Category Name:

Description:

SCREENSHOTS OF OUR DATABASE: ITEMS LIST

Welding

[<-- Back to categories list](#)

Torch:

A gas powered torch used to heat up or cut metal. | Price: 50.99 | In Stock: 11

[Read Reviews](#) | [Add to Cart](#) | (Admin: [Delete Item](#))

Welding Helmet:

A helmet that protects your eyes against harmful UV radiation. | Price: 20 | In Stock: 10

[Read Reviews](#) | [Add to Cart](#) | (Admin: [Delete Item](#))

(Admin) Add Item:

Item Name:

Description:

Price:

Stock Quantity:

SCREENSHOTS OF OUR DATABASE: SHOPPING CART

Cart:

[<-- Back to categories list](#)

Torch:

A gas powered torch used to heat up or cut metal. | Price: 50.99 | Quantity: 2

[Read Reviews](#) | [Add Another](#) | [Remove One](#) | [Remove All](#)

Portable Drill:

A portable power-drill that can be recharged. | Price: 25 | Quantity: 1

[Read Reviews](#) | [Add Another](#) | [Remove One](#) | [Remove All](#)

Welding Helmet:

A helmet that protects your eyes against harmful UV radiation. | Price: 20 | Quantity: 6

[Read Reviews](#) | [Add Another](#) | [Remove One](#) | [Remove All](#)

[Checkout](#) | [Clear Cart](#)

SCREENSHOTS OF OUR DATABASE: PREVIOUS ORDERS

Previous Orders:

[<-- Back to categories list](#)

Order number: 2 | Total Price: 126.98 | Date: 2027-11-20

Torch | Price: 50.99 | Quantity: 2 | [Read Reviews](#) | [Write A Review](#)

Portable Drill | Price: 25 | Quantity: 1 | [Read Reviews](#) | [Write A Review](#)

Order number: 3 | Total Price: 305.94 | Date: 2020-11-27

Torch | Price: 50.99 | Quantity: 6 | [Read Reviews](#) | [Write A Review](#)

Order number: 4 | Total Price: 379.95 | Date: 2020-11-27

Torch | Price: 50.99 | Quantity: 5 | [Read Reviews](#) | [Write A Review](#)

Portable Drill | Price: 25 | Quantity: 5 | [Read Reviews](#) | [Write A Review](#)

Order number: 5 | Total Price: 203.96 | Date: 2020-11-28

Torch | Price: 50.99 | Quantity: 4 | [Read Reviews](#) | [Write A Review](#)

SCREENSHOTS OF OUR DATABASE: ADDING A REVIEW

New Review:

Rating:

- ☐ 0
- ☐ 1
- ☐ 2
- ☐ 3
- ☐ 4
- ☐ 5

Enter review here...

Submit Review

[<-- Back to orders](#)

SCREENSHOTS OF OUR DATABASE: ITEM REVIEWS

Reviews:

[<-- Back to items](#)

Item: Portable Drill

Rating: 4/5

Reviewed by Admin: 4/5

Good quality and durability. Would buy again if it ever broke.

(Admin: Delete Review)

Reviewed by test: 5/5

Great tool! I've been using it for many projects and it's going strong.

(Admin: Delete Review)

