

Helen Pewther  
Sudeeksha Yadav  
COSC-280  
11/28/2022

### Test Document

**Note:** the question marks in the queries are a result of passing MySQL queries through Javascript. In practice, real values are being passed through the question marks.

-The real values are outlined under the test path/steps column

### Expected result from all successfully made queries:

-If the query returns data, that data is displayed to the GUI console

-If the query does not return data, but is successful, a success message should be displayed on the GUI.

### Insert

Sample query	Test path/steps	Expected result	Generated result	Successful
INSERT INTO users(first_name, last_name, email, uPassword, billing_address, date_registered) VALUES (?,?,?,?,?,?)	<b>Insert &gt; User Account:</b>  -F name: Jane -L name: Doe -email: jane.doe@gmail.com -password: janeDoe22 -address: 3700 O St NW, Washington DC 20774	Creates a new user account in the User table with the relevant data.  UID auto generated	-Relevant data inserted  -UID generated: 7	Yes
INSERT INTO orders(item_id, buyer_id, seller_id, order_date,	<b>Insert &gt; Order Invoice</b>  >Item ID: 2 >Buyer ID; 2	Creates a new order invoice in the database.	-Data inserted	Yes

order_cost) VALUES (?,?,?,?)	>Seller ID: 3 >Order Cost: \$13.99 >Order Date: 11/11/2022			
INSERT INTO orderstatus(trackin g_id, paid_status, shipping_status, shipping_provider) VALUES (?,?,?,?)	<b>Insert &gt; Order Invoice</b>  >Shipping provider: Fedex >Tracking number: 12345 >Shipping status: Order Received >Order paid: Paid	Creates a new entry in orderStatus table  Order_ID is auto-generate d	-Data successfull y inserted  -Order_ID generated: 4	Yes
INSERT INTO Products(product_ name, product_category, product_descriptio n) VALUES(?,?,?)	<b>Insert &gt; Product Type</b>  >Product Name: Supreme Nylon Jacket >Product Category: Nylon Jackets >Product Description: Jacket made of nylon from the streetwear brand, Supreme.	Error thrown from database.  The category does not yet exist.	-Error message from database.	Yes
INSERT INTO productCategories (category_name) VALUES(?)	<b>Insert &gt; Product Category</b>  >Category Name: Nylon Jackets	New entry made in TABLE productcateg ories	-New entry successfull y made	Yes
INSERT INTO Products(product_ name, product_category,	<b>Insert &gt; Product Type</b>  >Product Name: Supreme Nylon Jacket >Product Category: Nylon Jackets	New entry made in TABLE products	-New entry successfull y made	Yes

product_description) VALUES(?,?,?)	>Product Description: Jacket made of nylon from the streetwear brand, Supreme.			
INSERT INTO itemlistings(upload_er_id, product_id, item_image, upload_date, item_price, shipping_price, item_condition) VALUES (?,?,?,?,?,?,?)	<b>Insert &gt; Item Listing</b> >Seller ID : 10 >Product ID : 1 >Item condition: Brand new >Warehouse ID: 3 >Item asking price: 13.99 >Shipping price: 3.99	-Error message thrown from database.  -The warehouse ID does not belong to a known warehouse, therefore this query will not be successful.	-Error message from database/no response from database.	Yes
INSERT INTO itemlistings(upload_er_id, product_id, item_image, upload_date, item_price, shipping_price, item_condition) VALUES (?,?,?,?,?,?,?)	<b>Insert &gt; Item Listing</b> >Seller ID : 2 >Product ID : 1 >Item condition: Brand new >Warehouse ID: 3 >Item asking price: 13.99 >Shipping price: 3.99	-Successfully creates new entry in TABLE itemlistings	-New entry successfully made	Yes
INSERT INTO warehouses(city, the_state, country)	<b>Insert &gt; Warehouse</b> >City: Baltimore	-Creates a new entry in TABLE	-New entry successfully made	Yes

VALUES (?, ?, ?)	>State: Maryland >Country: USA	warehouse		
------------------	-----------------------------------	-----------	--	--

## Update

Sample query	Test path/steps	Expected result	Generated result	Successful
UPDATE users SET first_name = ? WHERE user_id = ?;	<b>UPDATE &gt; USER ACCOUNT</b>  >UID: 2 >F name: James	-Field first_name in TABLE users changes to JAMES	Field first_name in TABLE users changes to JAMES	Yes
UPDATE users SET last_name = ? WHERE user_id = ?;	<b>UPDATE &gt; USER ACCOUNT</b>  >UID: 2 >L name: Charles >Billing address:	-Field last_Name in TABLE users changes to Charles	Field last_Name in TABLE users changes to Charles	Yes
UPDATE users SET billing_address = ? WHERE user_id = ?;	<b>UPDATE &gt; USER ACCOUNT</b>  >UID: 2 >Billing address: 1350 Tondorf Rd, Washington DC 20774	Billing_addresses in TABLE users changes to specified address	Billing_address in TABLE users changes to specified address	Yes
UPDATE users SET email = ? WHERE user_id = ?;	<b>UPDATE &gt; USER ACCOUNT</b>  >UID: 2 >email: jane.doe@gmail.com	Changes email for entry with UID = 2	Changes email entry	Yes

?				
UPDATE orderstatus SET shipping_status= ? WHERE order_id = ?;	<b>UPDATE &gt; ORDER INVOICE</b>  >Order ID: 3 >shipping status: Order Recieved	Changes shipping_stat us in table orderstatus	Makes relevant changes	Yes
UPDATE orderstatus SET tracking_id= ? WHERE order_id = ?	<b>UPDATE &gt; ORDER INVOICE</b>  >Order ID: 3 >Tracking number: 123456	Changes tracking number in TABLE orderstatus	Makes relevant changes	Yes
UPDATE orderstatus SET shipping_provider = ? WHERE order_id = ?;	<b>UPDATE &gt; ORDER INVOICE</b>  >Order ID: 3 >shipping provider: UPS	Changes shipping_prov ider in TABLE orderstatus	Makes relevant changes	Yes
UPDATE orderstatus SET paid_status= ? WHERE order_id = ?;	<b>UPDATE &gt; ORDER INVOICE</b>  >Order ID: 3 >Payment status: declined	Changes paid_status in TABLE orderstatus	Makes relevant changes	yes
UPDATE warehouses SET city= ?, the_state=?, country=? WHERE warehouse_id = ?	<b>UPDATE &gt; WAREHOUSE</b>  >Warehouse ID : 2 >City: Buffalo >State: New York > Country: USA	Changes stored warehouse address in TABLE warehouses	Makes relevant changes	Makes relevant changes

## Read

Sample query	Test path/steps	Expected result	Generated result	Successful
SELECT * FROM productcategories;	<b>READ &gt; Product Type</b>	Displays column of ProductCategories table	Displays column of ProductCategories table	Yes
SELECT * FROM products where product_name = ? AND product_category = ?;	<b>READ &gt; Product Type</b>  >Product Name: Nike Air Force 1 >Product Category: Shoes	Displays row of Products table that have name = "Air Force 1" and category = "Shoes"	Displays row of Products table that have name = "Air Force 1" and category = "Shoes"  ("Nike Air Force 1", "Shoes", "The Air Force 1...")	Yes
SELECT * FROM products where product_name = ?;	<b>READ &gt; Product Type</b>  >Product Name: Nike Air Force 1	Displays rows of Products table that have name = "Nike Air Force 1"	Displays rows of Products table that have name = "Nike Air Force 1"	Yes
SELECT * FROM orders WHERE order_id=?	<b>READ &gt; Order Invoice</b>  >Order ID: 1	Displays row of Order table with ID = 1	Displays row of Order table with ID = 1  (1, 1, 1, 2, ('02/15/2022', 29.99),	Yes

SELECT * FROM users JOIN sellers ON users.user_id = sellers.seller_id WHERE users.user_id=?	<b>READ &gt; User Account</b>  >ID = 1 >show seller information	Displays a row of User data with seller information for user_id = 1	Displays a row of User data with seller information for user_id = 1  ("Jane", 'Doe', "jane.doe@gmail.com", "JaneDoe22", "809 Oberbrunner Stravenue", "9/25/2022", 1, 20, 20)	Yes
SELECT * FROM users WHERE users.user_id=?	<b>READ &gt; User Account</b>  >ID = 1	Displays a row of User data for user_id = 1	Displays a row of User data for user_id = 1  ("Jane", 'Doe', "jane.doe@gmail.com", "JaneDoe22", "809 Oberbrunner Stravenue", "9/25/2022")	Yes
SELECT * FROM users WHERE (first_name = ? AND last_name= ?);	<b>Read &gt; User Account</b>  >First Name: Jane >Last Name: Doe	Displays rows of User data with first_name = "Jane" and last_name = "Doe"	Displays rows of User data with first_name = "Jane" and last_name = "Doe"  ("Jane", 'Doe', "jane.doe@gmail.com", "JaneDoe22", "809 Oberbrunner	Yes

			Stravenue", "9/25/2022")	
SELECT * FROM users JOIN sellers WHERE (users.first_name = ? AND users.last_name= ?)	<b>Read &gt; User Account</b>  >First Name: Jane >Last Name: Doe >show seller information	Displays rows of User data with first_name = "Jane" and last_name = "Doe" with seller information	Displays rows of User data with first_name = "Jane" and last_name = "Doe" with seller information  ("Jane", 'Doe', "jane.doe@gmail.com", "JaneDoe22", "809 Oberbrunner Stravenue", "9/25/2022")	Yes
SELECT * FROM users WHERE (users.first_name LIKE ?% AND users.last_name LIKE ?%)	<b>Read &gt; User Account</b>  >First Initial: R >Last Initial: R	Displays rows of User data with first_name starts with "R" and last_name starts with "R"	Displays rows of User data with first_name starts with "R" and last_name starts with "R"  ("Range", "Rover", "range.rover@gmail.c om", "rr22", "Blue Wing terrace, MD 20774", "11/11/2022")	Yes
SELECT * FROM users JOIN sellers WHERE (users.first_name LIKE ?% AND	<b>Read &gt; User Account</b>  >First Initial: R >Last Initial: R	Displays rows of User data with first_name starts with "R" and last_name starts with "R" with seller info	Displays rows of User data with first_name starts with "R" and last_name starts with "R"	Yes



users.last_name LIKE ?%)	>show seller information		("Range", "Rover", "range.rover@gmail.com", "rr22", "Blue Wing terrace, MD 20774", "11/11/2022", 3, 20, 0)	
SELECT * FROM itemlistings WHERE item_id = ?	<b>Read &gt; Item Listing</b>  >Item ID = 1	Displays row of ItemListing table where item_id = 1	Displays row of ItemListing table where item_id = 1  (1, 1, null, ('01/05/2010'), 5.99, 1.99, "gently used")	Yes
SELECT * FROM warehouses WHERE warehouse_id = ?	<b>Read &gt; Warehouse Info</b>  >Warehouse ID = 1	Displays row of Warehouse table where warehouse_id = 1	Displays row of Warehouse table where warehouse_id = 1  ("Washington", "DC", "USA")	Yes

## Delete

Sample query	Test path/steps	Expected result	Generated result	Successful
DELETE FROM itemlistings WHERE item_id = ?	<b>DELETE &gt; Item Listing</b>  >Item ID: 2	Sends delete query to TABLE itemlistings	Item listing with ID = 2 is deleted from the database	Yes

DELETE FROM Users WHERE user_id = ?	<b>DELETE &gt; User Account</b> >User ID: 1	Sends delete query to TABLE Users	User with ID = 1 is deleted from the database	Yes
DELETE FROM orders WHERE order_id = ?	<b>DELETE &gt; Order Invoice</b> >Order ID = 1	Sends delete query to Orders table for Order ID = 1	Order row of ID = 1 is deleted from the database	Yes
DELETE FROM productcategories WHERE category_name = ?	<b>DELETE &gt; productcategories</b> >Product Category = "Dresses"	Sends delete query to ProductCategories table for category_name = "Dresses"	ProductCategories row where name = "Dresses" is deleted from the database	Yes
DELETE FROM products WHERE product_id = ?	<b>DELETE &gt; Product Category</b> >Product ID = 1	Sends delete query to Products table for Product ID = 1	Products row of ID = 1 is deleted from the database	Yes
DELETE FROM warehouses WHERE warehouse_id = ?	<b>DELETE &gt; Warehouse Info</b> >Warehouse ID = 1	Sends delete query to Warehouse table for Warehouse ID = 1	Warehouse row of ID = 1 is deleted from the database	Yes

### Custom Query

Sample query	Test path/steps	Expected result	Generated result	Successful
SELECT * FROM	<b>Query &gt; Custom</b>	All rows from user	Shows all data from	Yes

users;	<b>query</b>	table are displayed in the UI console.	table user.	
SELECT MIN(order_cost) FROM Orders;	<b>Query &gt; Custom query</b>	Displays minimum order cost in TABLE orders	{"MIN(order_cost)": "10.99"}	Yes
SELECT COUNT(user_id) FROM Users WHERE users.first_Name LIKE "Jane";	<b>Query &gt; Custom query</b>	Displays number of users named Jane in userbase	{"COUNT(user_id)": 2}	Yes

### Non-functional features:

#### 1. READ > Order Invoice > Advanced Search options

```
SELECT * FROM Orders
WHERE order_cost > mincost AND order_cost < maxcost;
```

(queries to complete these operations are functional - just ran out of time and couldn't connect the queries up to the UI before the deadline).