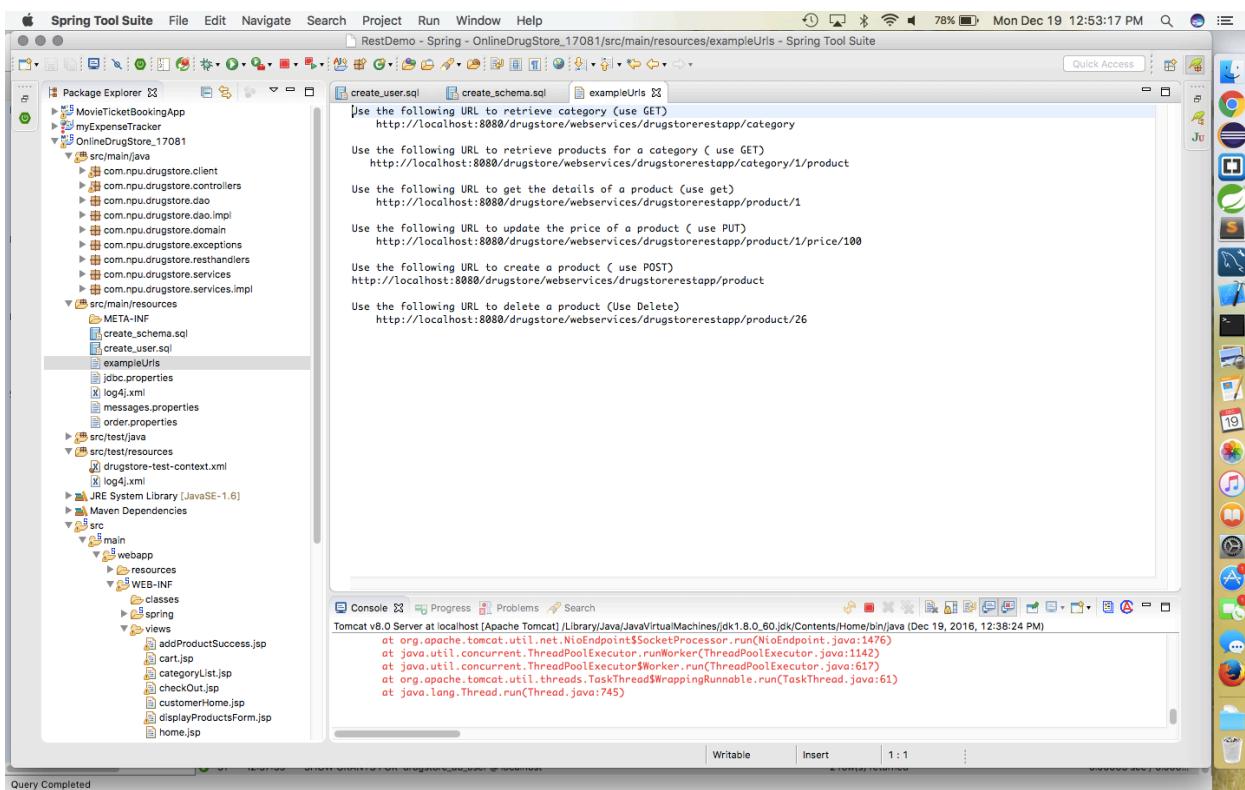
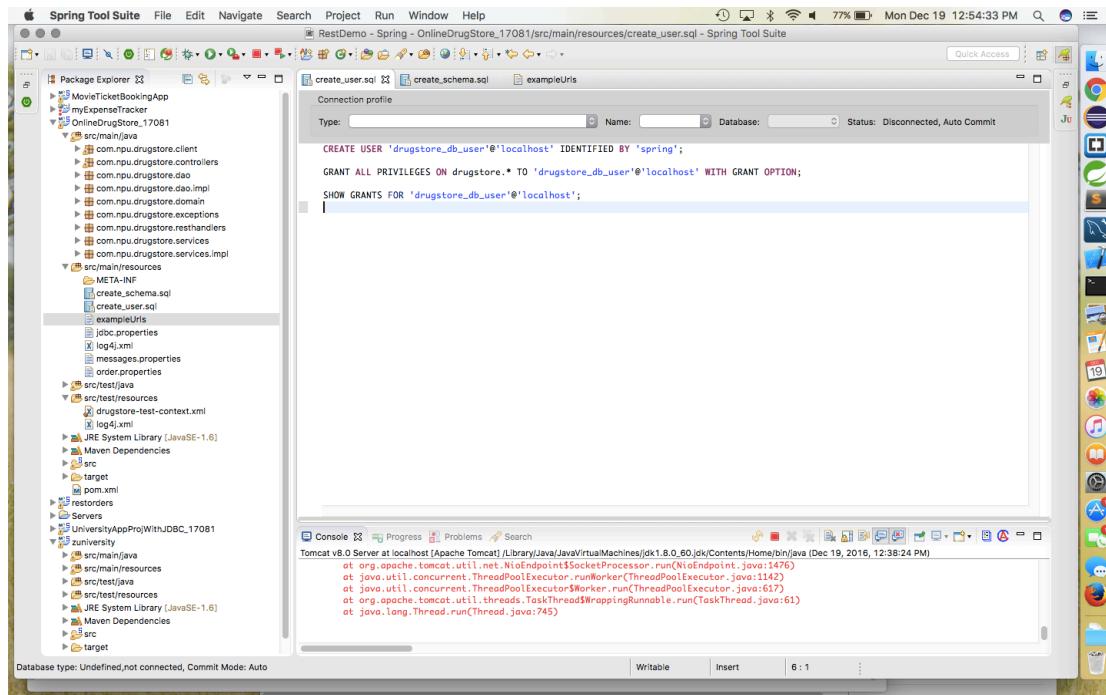


Abstract

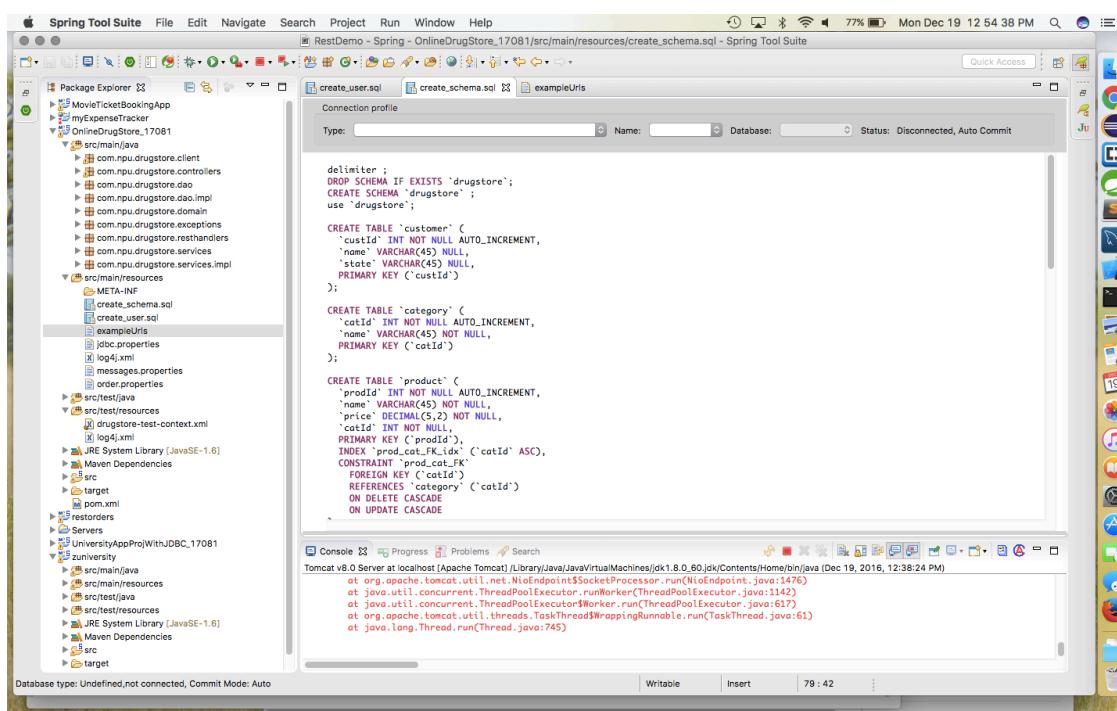
Online Drug Store Application is a web-based application for purchasing any drug. This application allows customers to select the category of the drug from categories list and with the product name with quantity needed and can purchase easily. Users just need to sign in with their names and can purchase the drugs.

Screen shots:

Database: Create User



Create schema and tables:



MySQLWorkbench

File Edit View Query Database Server Tools Scripting Help

MySQL Workbench

hh x kj x

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

SCHEMAS

- drugstore
 - Tables
 - category
 - customer
 - inventory
 - orderitem
 - orders
 - product
 - Views
 - Stored Procedures
 - Functions
- sys

Query 1 SQL File 1* category customer

1 • SELECT * FROM drugstore.customer;

Result Grid Filter Rows: Search Edit: Export/Import:

custid	name	state
1	Susan	MI
2	Joy	CA
3	Garam	CA
4	Lucas	LA
5	Henry	TX
HULL	HULL	HULL

Action Output

Time	Action	Response	Duration / Fetch Time
12:22:48	SELECT * FROM drugstore.category LIMIT 0, 10	7 row(s) returned	0.00047 sec / 0.000...
12:22:57	SELECT * FROM drugstore.customer LIMIT 0, 10	5 row(s) returned	0.0022 sec / 0.0001...

customer 1

Apply Revert

Query Completed

The screenshot shows the MySQL Workbench interface with the 'customer' table selected. The results grid displays five rows of data: Susan (MI), Joy (CA), Garam (CA), Lucas (LA), and Henry (TX). The action output table shows two queries: one for the category table and one for the customer table, both returning the requested number of rows within the specified time.

MySQLWorkbench

File Edit View Query Database Server Tools Scripting Help

MySQL Workbench

hh x kj x

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

SCHEMAS

- drugstore
 - Tables
 - category
 - customer
 - inventory
 - orderitem
 - orders
 - product
 - Views
 - Stored Procedures
 - Functions
- sys

Query 1 SQL File 1* category

1 • SELECT * FROM drugstore.category;

Result Grid Filter Rows: Search Edit: Export/Import:

catid	name
1	Inhalants
2	Cannabinoids
3	Depressants
4	Opioids & Morphine
5	Anabolic Steroids
6	Hallucinogens
7	Prescription
HULL	HULL

Action Output

Time	Action	Response	Duration / Fetch Time
12:22:48	SELECT * FROM drugstore.category LIMIT 0, 10	7 row(s) returned	0.00047 sec / 0.000...

category 1

Apply Revert

Query Completed

The screenshot shows the MySQL Workbench interface with the 'category' table selected. The results grid displays seven categories: Inhalants, Cannabinoids, Depressants, Opioids & Morphine, Anabolic Steroids, Hallucinogens, and Prescription. The action output table shows one query for the category table, returning the requested number of rows within the specified time.

MySQLWorkbench

File Edit View Query Database Server Tools Scripting Help

Mon Dec 19 12:23:22 PM

hh x kj x

MySQL Workbench

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

SCHEMAS

- drugstore
- sys

Tables

- category
- customer
- inventory
- orderitem
- orders
- product

Views

Stored Procedures

Functions

Result Grid

Filter Rows:

Search

Edit: Export/Import:

prodid	name	price	catid
1	Aerosol sprays	40.00	1
2	Oxycodone	30.00	4
3	Glue	30.00	1
4	Durabolin	25.00	5

product 1

Action Output

Time	Action	Response	Duration / Fetch Time
12/22/57	SELECT * FROM drugstore.customer LIMIT 0, 10	5 row(s) returned	0.0022 sec / 0.0001...
12/23/02	SELECT * FROM drugstore.inventory LIMIT 0, 10	3 row(s) returned	0.0052 sec / 0.000...
12/23/08	SELECT * FROM drugstore.orderitem LIMIT 0, 10	0 row(s) returned	0.0084 sec / 0.000...
12/23/14	SELECT * FROM drugstore.orders LIMIT 0, 10	0 row(s) returned	0.0048 sec / 0.000...
12/23/17	SELECT * FROM drugstore.product LIMIT 0, 10	4 row(s) returned	0.0015 sec / 0.0001...

Query Completed

The screenshot shows the MySQL Workbench interface with the 'product' table selected. The results grid displays four rows of data: prodid 1 (name 'Aerosol sprays', price 40.00, catid 1), prodid 2 ('Oxycodone', 30.00, 4), prodid 3 ('Glue', 30.00, 1), and prodid 4 ('Durabolin', 25.00, 5). Below the results grid is an 'Action Output' table showing six log entries corresponding to the SELECT statements executed.

MySQLWorkbench

File Edit View Query Database Server Tools Scripting Help

Mon Dec 19 12:23:05 PM

hh x kj x

MySQL Workbench

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

SCHEMAS

- drugstore
- sys

Tables

- category
- customer
- inventory
- orderitem
- orders
- product

Views

Stored Procedures

Functions

Result Grid

Filter Rows:

Search

Edit: Export/Import:

invId	prodId	initialCount	availableCount	soldCount	sellingDetails
1	3	30	30	30	NULL
2	2	20	20	20	NULL
3	1	30	30	30	NULL

inventory 1

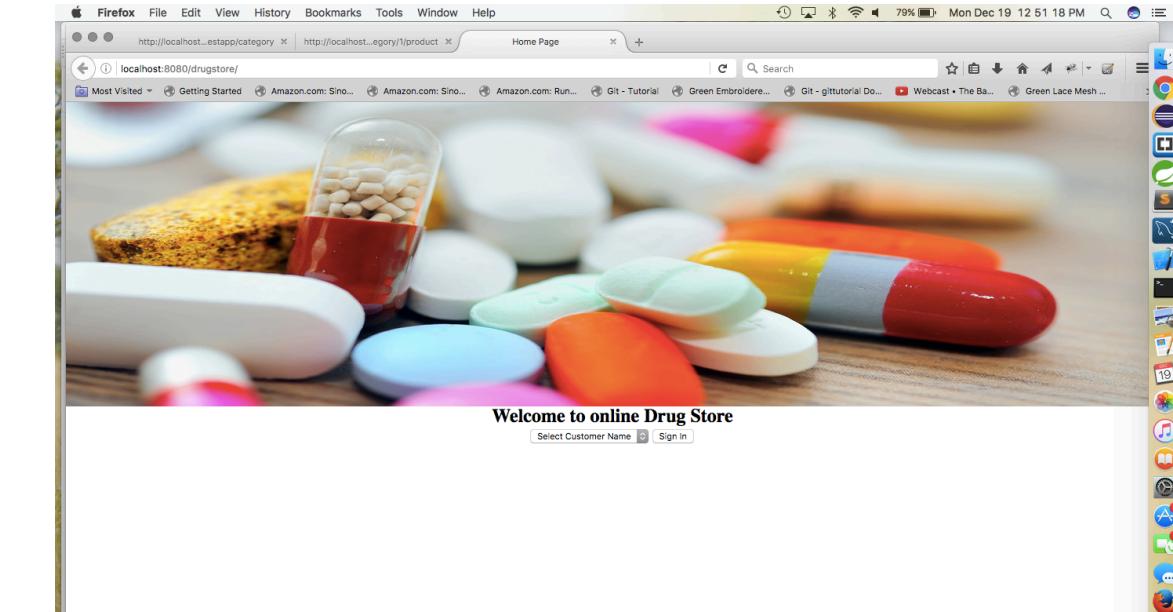
Action Output

Time	Action	Response	Duration / Fetch Time
12/22/48	SELECT * FROM drugstore.category LIMIT 0, 10	7 row(s) returned	0.00047 sec / 0.000...
12/22/57	SELECT * FROM drugstore.customer LIMIT 0, 10	5 row(s) returned	0.0022 sec / 0.0001...
12/23/02	SELECT * FROM drugstore.inventory LIMIT 0, 10	3 row(s) returned	0.00052 sec / 0.000...

Query Completed

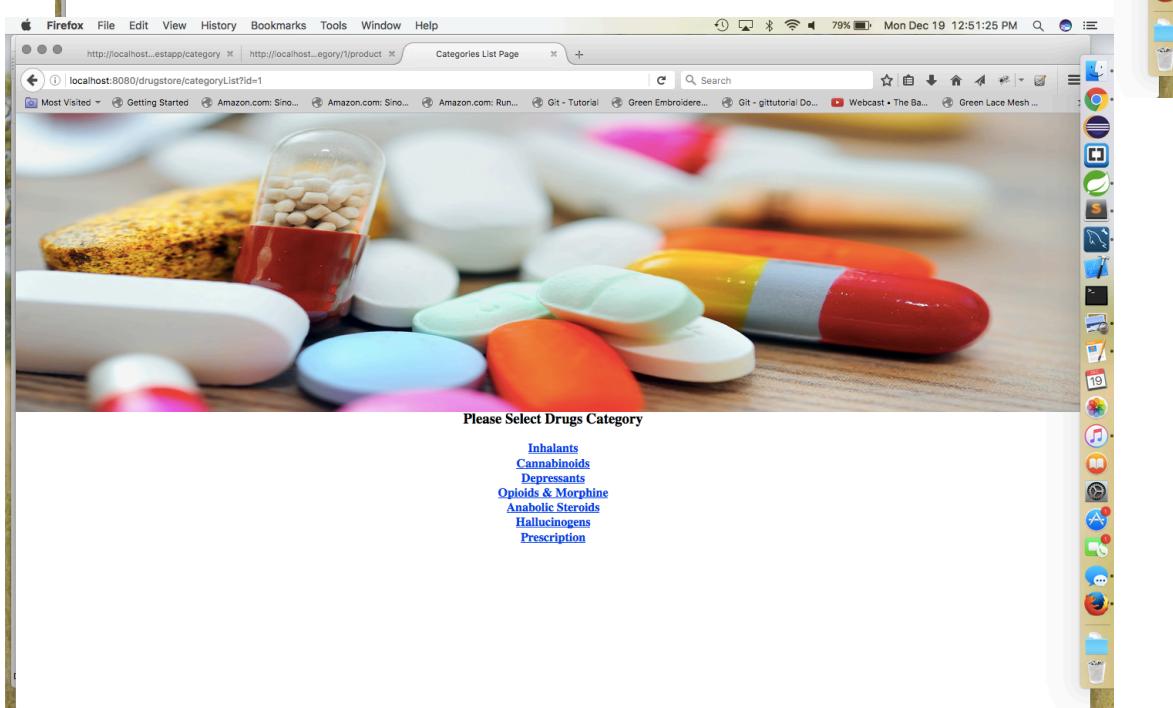
The screenshot shows the MySQL Workbench interface with the 'inventory' table selected. The results grid displays three rows of data: invId 1 (prodId 3, initialCount 30, availableCount 30, soldCount 30, sellingDetails NULL), invId 2 (prodId 2, 20, 20, 20, NULL), and invId 3 (prodId 1, 30, 30, 30, NULL). Below the results grid is an 'Action Output' table showing three log entries corresponding to the SELECT statements executed.

Application:



Welcome to online Drug Store

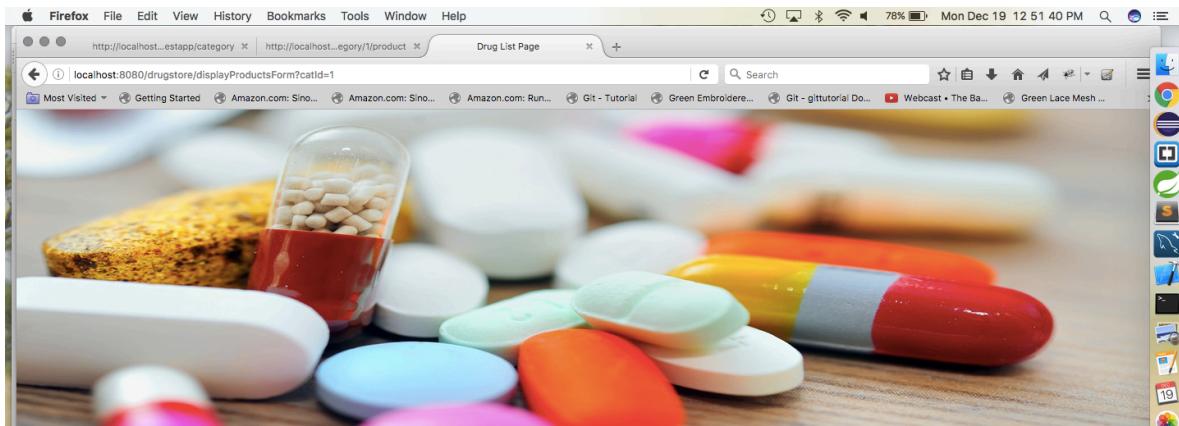
Select Customer Name



Categories List Page

Please Select Drugs Category

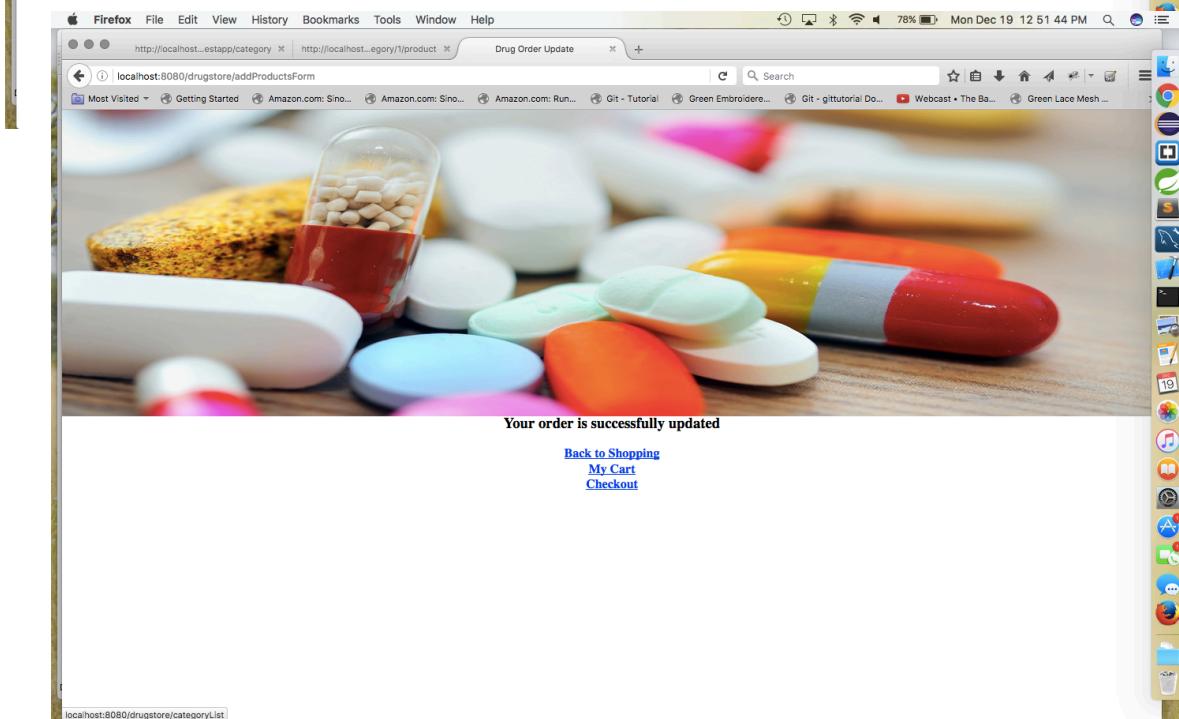
- [Inhalants](#)
- [Cannabinoids](#)
- [Depressants](#)
- [Opioids & Morphine](#)
- [Anabolic Steroids](#)
- [Hallucinogens](#)
- [Prescription](#)



Please select the drugss and quantity

drug Name	Drug Price	Quantity
Aerosol sprays	40.0	2
Glues	30.0	0

[Add to cart](#)



Your order is successfully updated

[Back to Shopping](#)
[My Cart](#)
[Checkout](#)

<localhost:8080/drugstore/categoryList>

Firefox File Edit View History Bookmarks Tools Window Help

78% Mon Dec 19 12:51:49 PM

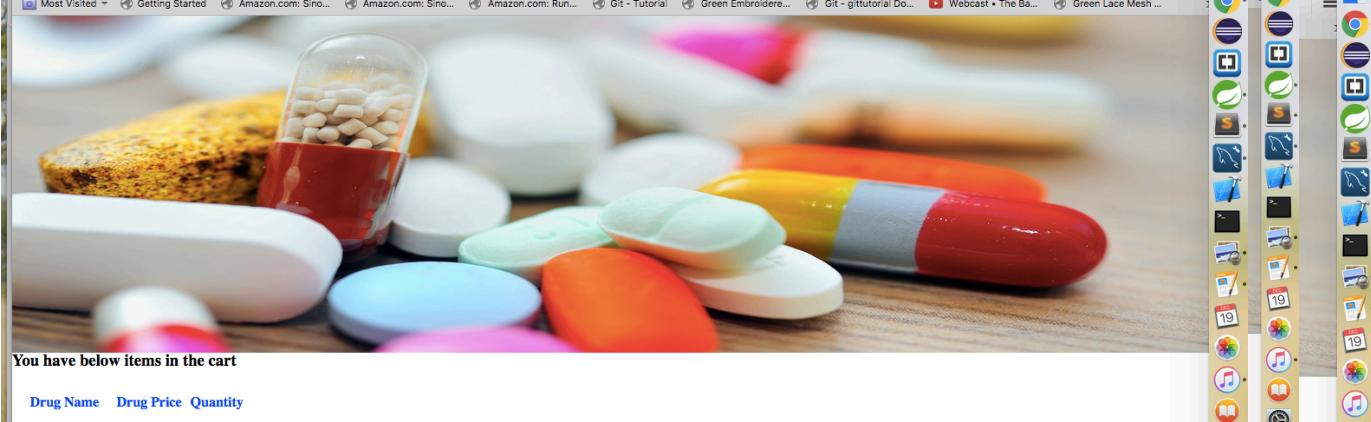
localhost:8080/drugstore/cart http://localhost:8080/drugstore/cart Your Cart

Most Visited Getting Started Amazon.com: Sino... Amazon.com: Sino... Amazon.com: Run... Git - Tutorial Green Embroidere... Git - gittutorial Do... Webcast • The Ba... Green Lace Mesh ...

You have below items in the cart

Drug Name	Drug Price	Quantity
Aerosol sprays	40.0	2

[Back to Shopping](#)
[Check Out](#)



Database table explanation:

There are

- Customer table,
This table consists of customer details like customer name, state and customer id.
- Category table,
This table consists of a list of categories of drugs like (Inhalants, Anabolic Steroids,etc.)
- Product table,
This table consists of all the drug products under the selected category- category id, name of the product, product id, price
- Inventory table,
This table consists of all the product details with the initial count, available count and sold count.
- Order item table,
This table consists of the order made by the customer with the product id and quantity.
- Orders table,
This table consists of the order made by who and how much? like customer id and total purchase price.

Example URLs:

Use the following URL to retrieve category (use GET)

`http://localhost:8080/drugstore/webservices/drugstorerestapp/category`

Use the following URL to retrieve products for a category (use GET)

`http://localhost:8080/drugstore/webservices/drugstorerestapp/category/1/product`

Use the following URL to get the details of a product (use get)

`http://localhost:8080/drugstore/webservices/drugstorerestapp/product/1`

Sample Data:

```
delimiter ;
DROP SCHEMA IF EXISTS `drugstore`;
CREATE SCHEMA `drugstore` ;
use `drugstore`;

CREATE TABLE `customer` (
  `custId` INT NOT NULL AUTO_INCREMENT,
  `name` VARCHAR(45) NULL,
  `state` VARCHAR(45) NULL,
  PRIMARY KEY (`custId`)
);

CREATE TABLE `category` (
  `catId` INT NOT NULL AUTO_INCREMENT,
  `name` VARCHAR(45) NOT NULL,
  PRIMARY KEY (`catId`)
);

CREATE TABLE `product` (
  `prodId` INT NOT NULL AUTO_INCREMENT,
  `name` VARCHAR(45) NOT NULL,
  `price` DECIMAL(5,2) NOT NULL,
  `catId` INT NOT NULL,
  PRIMARY KEY (`prodId`),
  INDEX `prod_cat_FK_idx` (`catId` ASC),
  CONSTRAINT `prod_cat_FK`
    FOREIGN KEY (`catId`)
    REFERENCES `category` (`catId`)
    ON DELETE CASCADE
    ON UPDATE CASCADE
);
CREATE TABLE `orders` (
  `orderId` INT NOT NULL AUTO_INCREMENT,
  `custId` INT NOT NULL,
  `subtotal` DECIMAL(11,3) NULL,
  `tax` DECIMAL(6,2) NULL,
  `total` DECIMAL(15,3) NULL,
  PRIMARY KEY (`orderId`),
  INDEX `order_cust_FK_idx` (`custId` ASC),
  CONSTRAINT `order_cust_FK`
    FOREIGN KEY (`custId`)
    REFERENCES `customer` (`custId`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION
);
CREATE TABLE `orderitem` (
  `orderItemId` int(11) NOT NULL AUTO_INCREMENT,
  `orderId` int(11) NOT NULL,
  `prodId` int(11) NOT NULL,
```

```

`quantity` int(11) NOT NULL,
PRIMARY KEY (`orderItemId`),
KEY `orderItem_product_FK_idx` (`prodId`),
KEY `orderItem_order_FK` (`orderId`),
CONSTRAINT `orderItem_order_FK` FOREIGN KEY (`orderId`) REFERENCES `orders`(`orderId`) ON DELETE NO ACTION ON UPDATE NO ACTION,
CONSTRAINT `orderItem_product_FK` FOREIGN KEY (`prodId`) REFERENCES `product`(`prodId`) ON DELETE NO ACTION ON UPDATE NO ACTION
);

```

```

CREATE TABLE `inventory` (
`invId` INT NOT NULL AUTO_INCREMENT,
`prodId` INT NOT NULL,
`initialCount` INT NOT NULL,
`availableCount` INT NOT NULL,
`soldCount` INT NOT NULL,
`sellingDetails` VARCHAR(45) NULL,
PRIMARY KEY (`invId`),
UNIQUE INDEX `prodId_UNIQUE` (`prodId` ASC),
CONSTRAINT `Inventory_Prod_FK` FOREIGN KEY (`prodId`) REFERENCES `product`(`prodId`) ON DELETE NO ACTION ON UPDATE NO ACTION
);

```

```

INSERT INTO `drugstore`.`category` (`name`) VALUES ('Inhalants');
INSERT INTO `drugstore`.`category` (`name`) VALUES ('Cannabinoids');
INSERT INTO `drugstore`.`category` (`name`) VALUES ('Depressants');
INSERT INTO `drugstore`.`category` (`name`) VALUES ('Opioids & Morphine');
INSERT INTO `drugstore`.`category` (`name`) VALUES ('Anabolic Steroids');
INSERT INTO `drugstore`.`category` (`name`) VALUES ('Hallucinogens');
INSERT INTO `drugstore`.`category` (`name`) VALUES ('Prescription');

INSERT INTO `drugstore`.`customer` (`name`, `state`) VALUES ('Susan', 'MI');
INSERT INTO `drugstore`.`customer` (`name`, `state`) VALUES ('Joy', 'CA');
INSERT INTO `drugstore`.`customer` (`name`, `state`) VALUES ('Garam', 'CA');
INSERT INTO `drugstore`.`customer` (`name`, `state`) VALUES ('Lucas', 'LA');
INSERT INTO `drugstore`.`customer` (`name`, `state`) VALUES ('Henry', 'TX');

INSERT INTO `drugstore`.`product` (`name`, `price`, `catId`) VALUES ('Aerosol sprays\n', '40', '1');
INSERT INTO `drugstore`.`product` (`name`, `price`, `catId`) VALUES ('Oxycodone', '30', '4');
INSERT INTO `drugstore`.`product` (`name`, `price`, `catId`) VALUES ('Glues', '30', '1');
INSERT INTO `drugstore`.`product` (`name`, `price`, `catId`) VALUES ('Durabolin', '25', '5');

INSERT INTO `drugstore`.`inventory` (`prodId`, `initialCount`, `availableCount`, `soldCount`) VALUES ('3', '30', '30', '30');
INSERT INTO `drugstore`.`inventory` (`prodId`, `initialCount`, `availableCount`, `soldCount`) VALUES ('2', '20', '20', '20');

```

```
INSERT INTO `drugstore`.`inventory` (`prodId`, `initialCount`, `availableCount`, `soldCount`)
VALUES ('1', '30', '30', '30');
```

