# bamazon\_app

#1: Customer View

Create a MySQL Database called bamazon.

Then create a Table inside of that database called products.

DROP DATABASE IF EXISTS bamazon;

CREATE DATABASE bamazon;

USE bamazon;

CREATE TABLE products (

id INTEGER(10) AUTO\_INCREMENT,

product\_name VARCHAR(200) NOT NULL,

department\_name VARCHAR(200) NOT NULL,

price int(20) NOT NULL,

stock\_quantity int(20) NOT NULL,

PRIMARY KEY(id)

);

Populate this database with around 10 different products. (i.e. Insert "mock" data rows into this database and table).

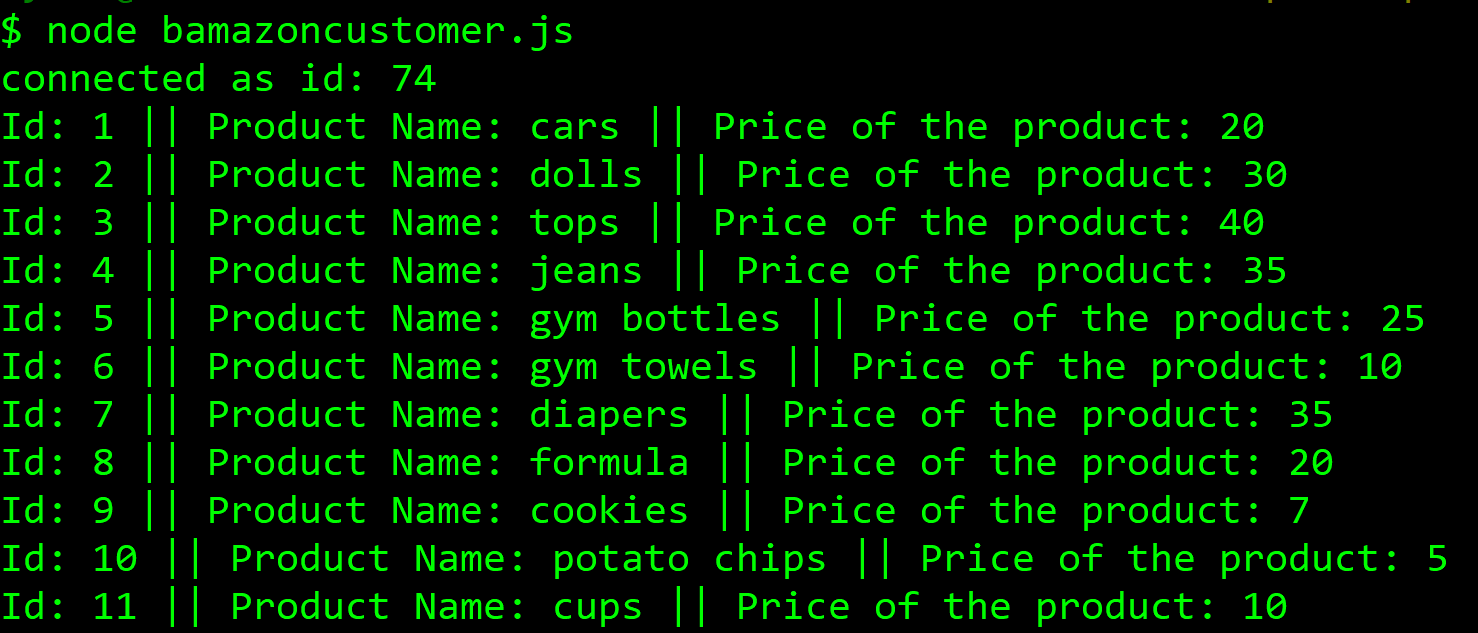
-Create a csv file & use import wizard to import

Then create a Node application called bamazonCustomer.js.

Running this application will first display all of the items available for sale.

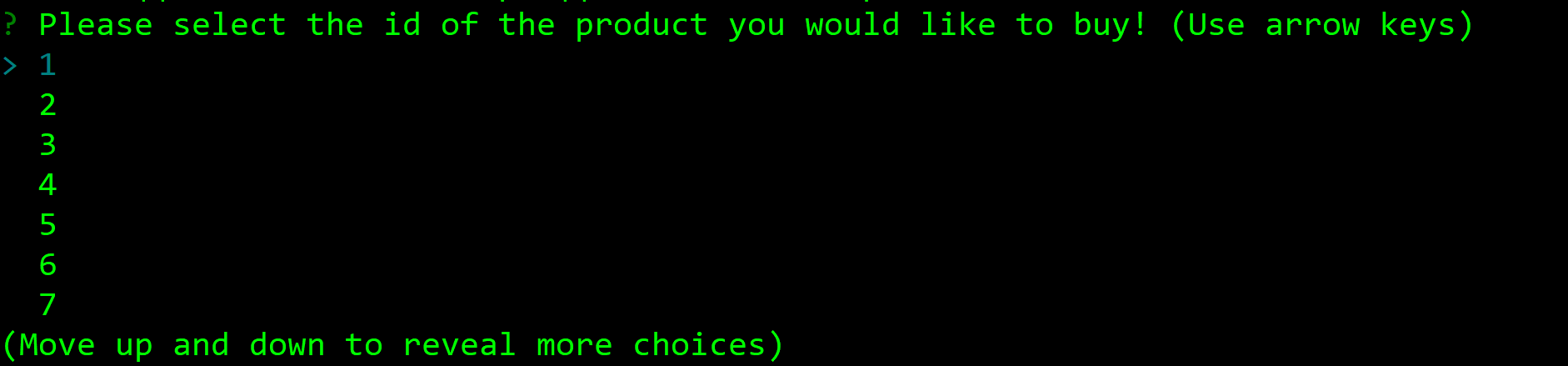
Include the ids, names, and prices of products for sale.

showAll()

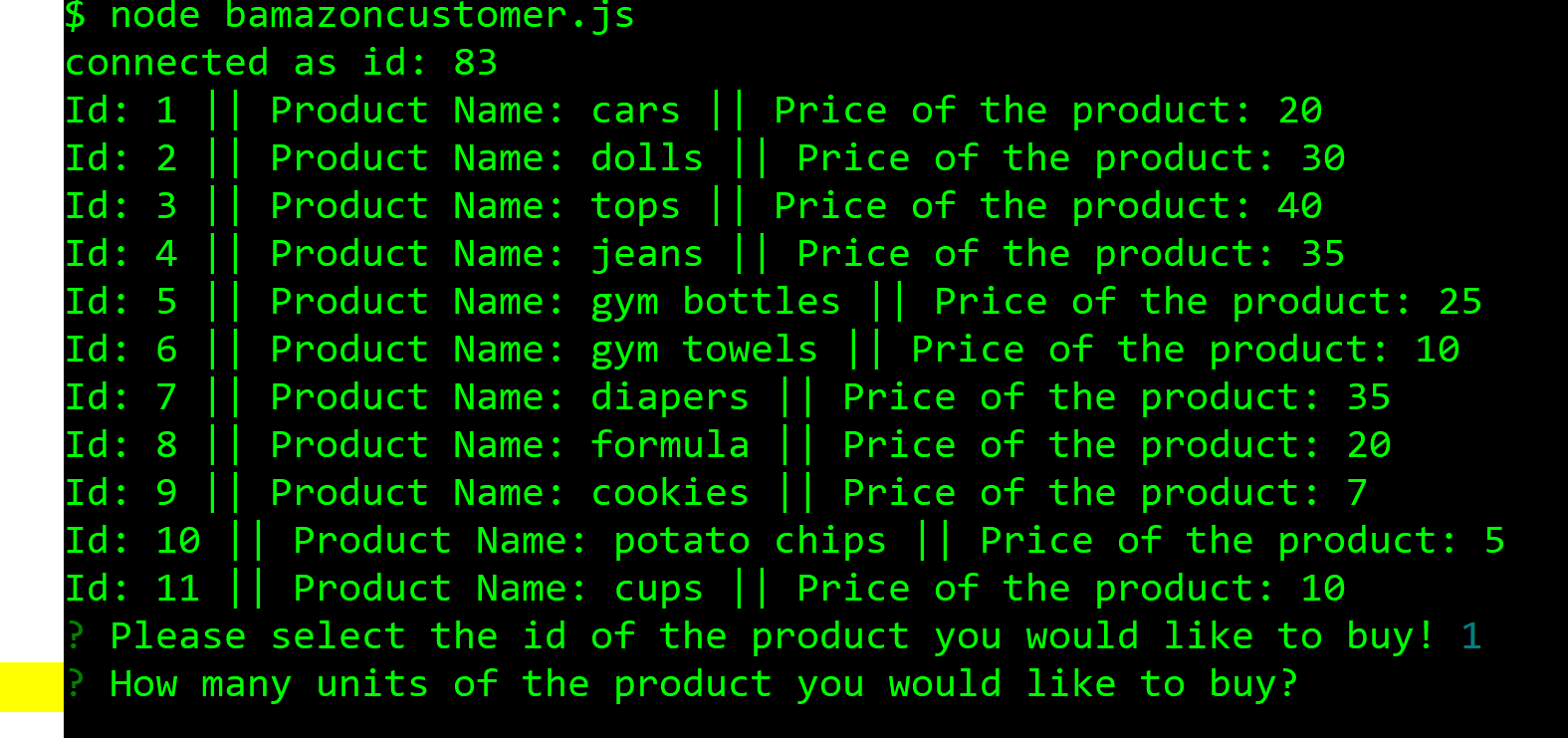


The app should then prompt users with two messages.

The first should ask them the ID of the product they would like to buy.

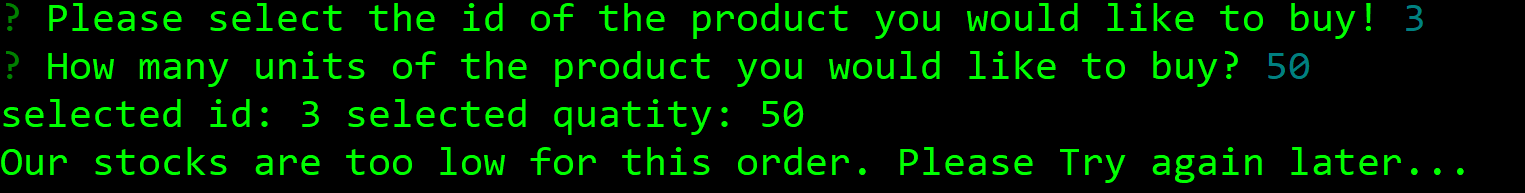


The second message should ask how many units of the product they would like to buy.



Once the customer has placed the order, your application should check if your store has enough of the product to meet the customer's request.

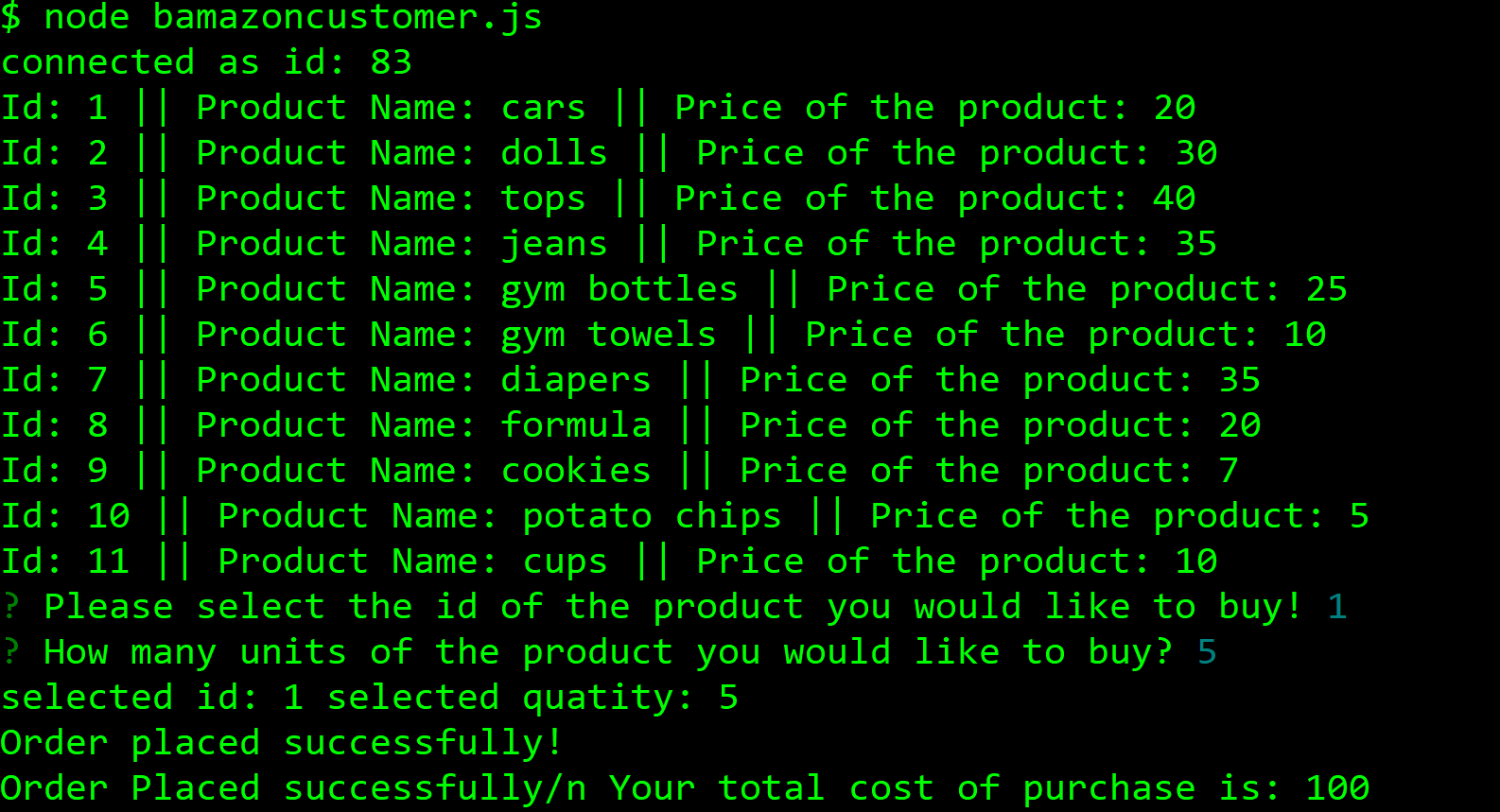
If not, the app should log a phrase like Insufficient quantity!, and then prevent the order from going through.



However, if your store does have enough of the product, you should fulfill the customer's order.

This means updating the SQL database to reflect the remaining quantity.

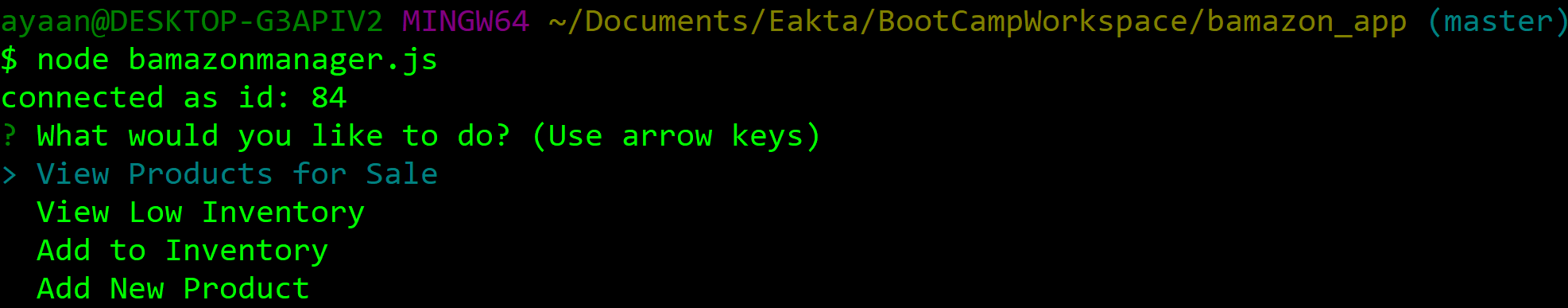
Once the update goes through, show the customer the total cost of their purchase.



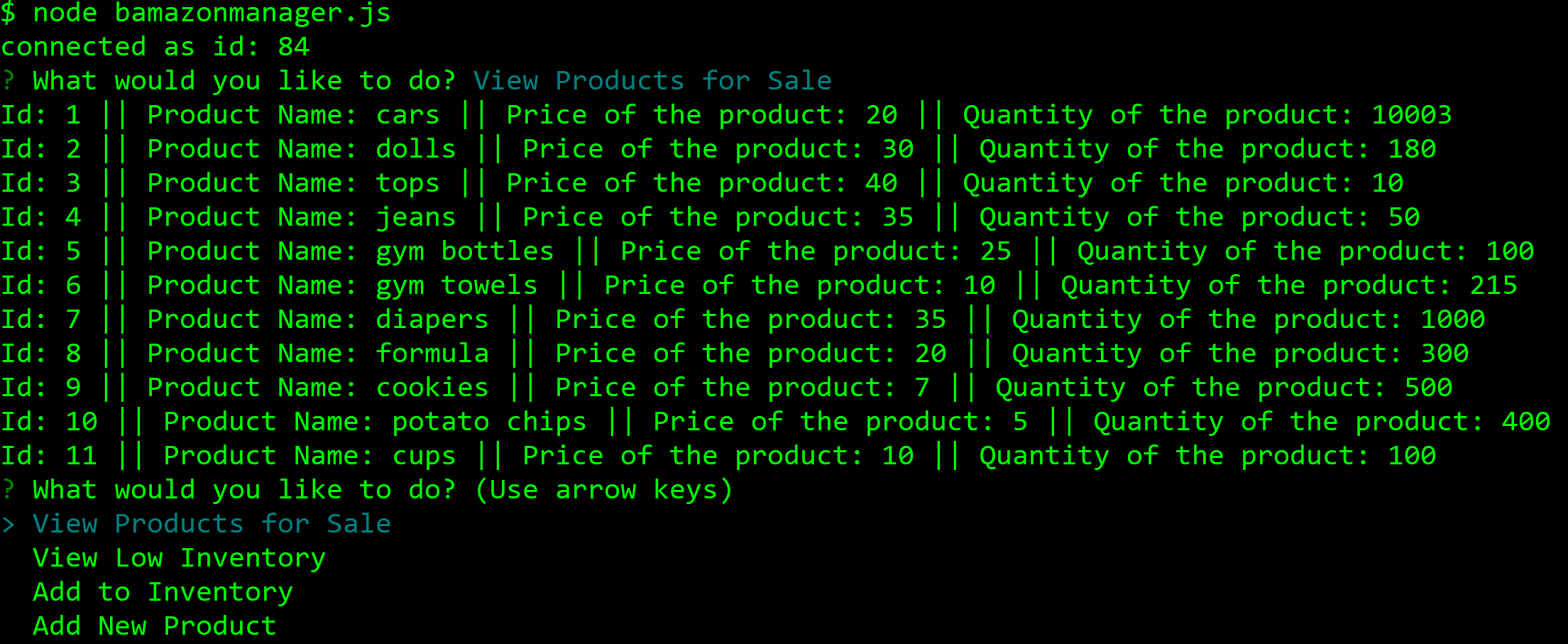
#2 Manager’s View

Create a new Node application called bamazonManager.js. Running this application will:

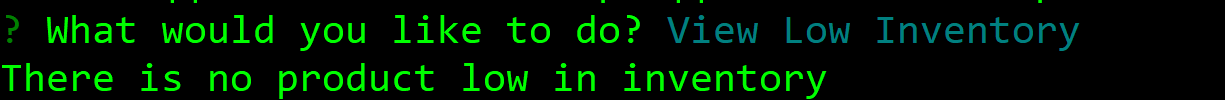
* List a set of menu options:
* View Products for Sale
* View Low Inventory
* Add to Inventory
* Add New Product



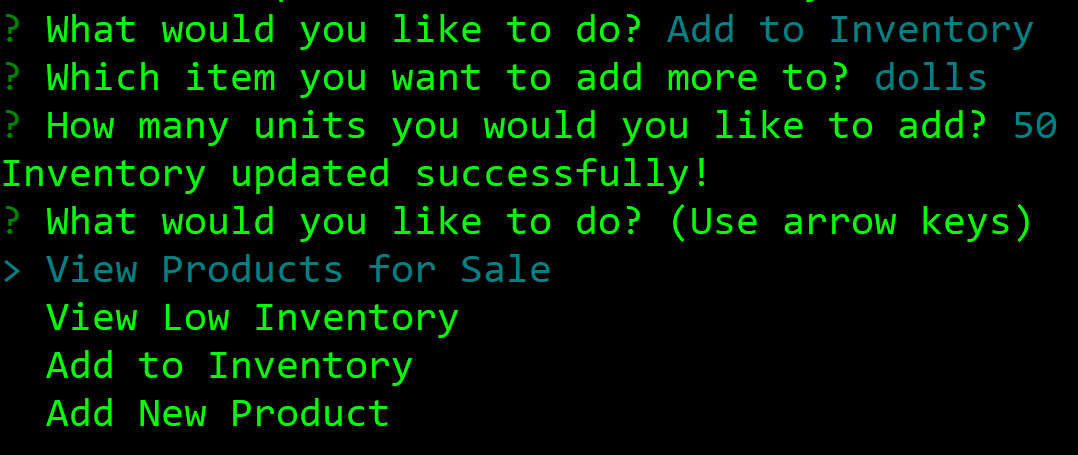
* If a manager selects View Products for Sale, the app should list every available item: the item IDs, names, prices, and quantities.



* If a manager selects View Low Inventory, then it should list all items with an inventory count lower than five.



* If a manager selects Add to Inventory, your app should display a prompt that will let the manager "add more" of any item currently in the store.



* If a manager selects Add New Product, it should allow the manager to add a completely new product to the store.

