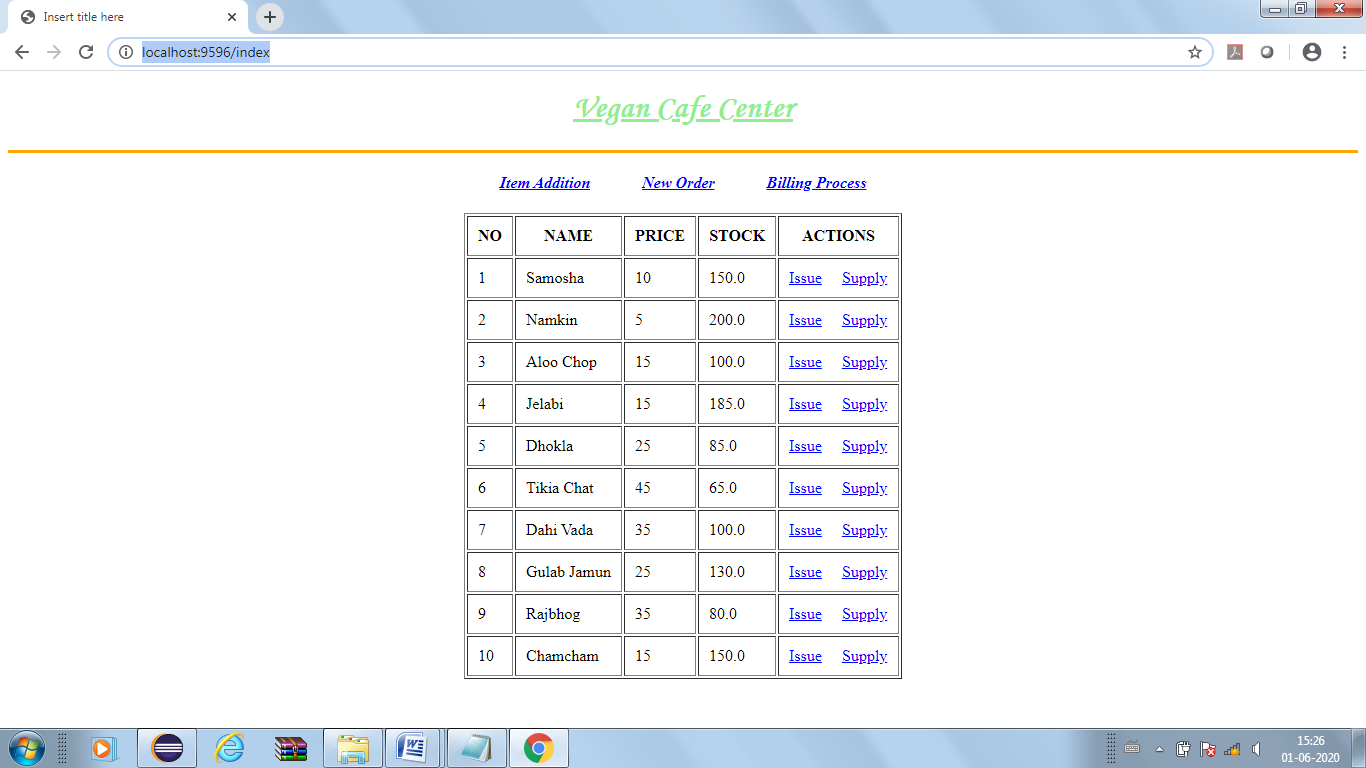
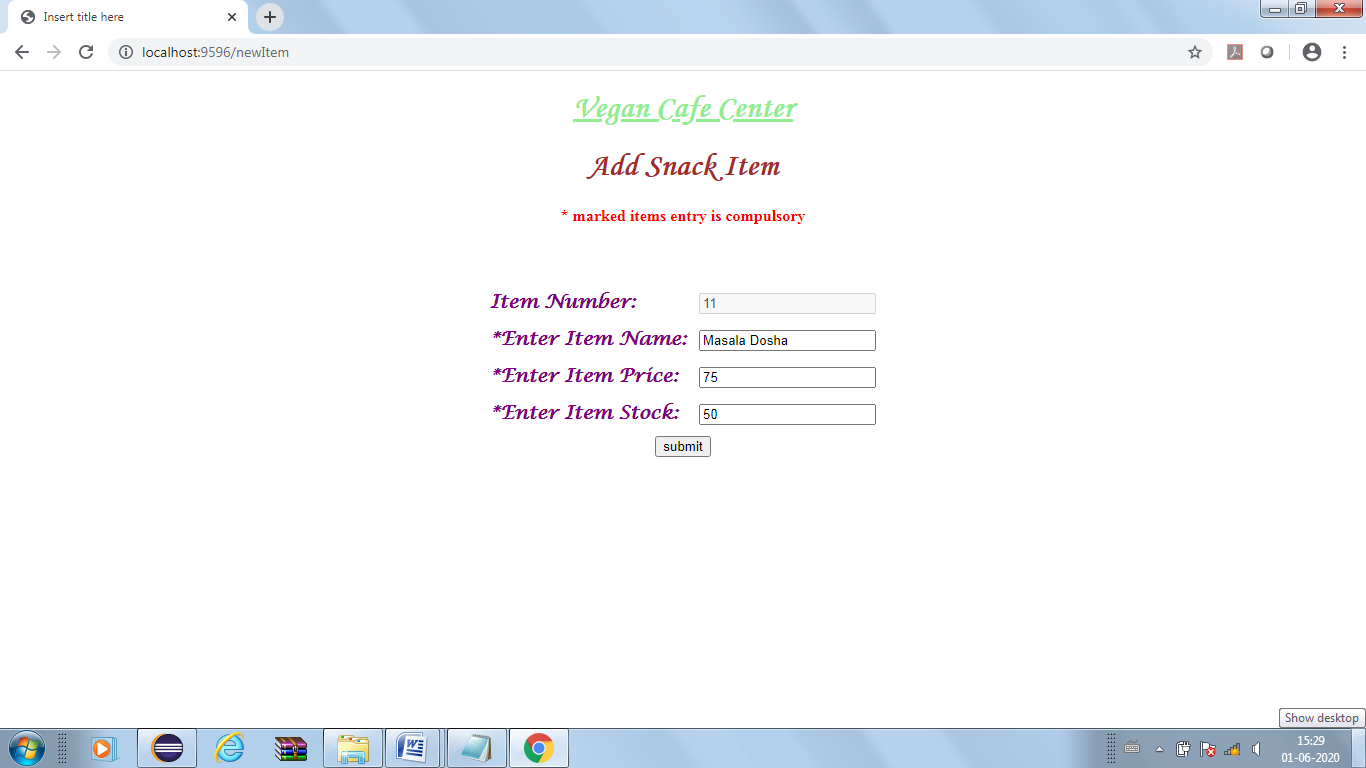
***Vegan-Cafe***, sales pure vegetarian various snack items which includes both salty and sweet. The shop opens from 8:00 am to 10:00 pm. There are 3 counters to handle customers. The counters are handling the supply of snack items from in-house kitchen to the shop, add any new snack item in the snacks list, accept customer orders and generate bills.

You need to develop a web based application covering of above features on SpringBoot plus String Data JPA. Database can be MySQL or Oracle.

If you type *http://localhost:<port \_no>/index* the following will be appeared:-



If you click “*Item Addition*” the output:-



*Item number* will be auto generated and non editable, after clicking submit button the output:-



If you click *“New Order”* the output is as follows:-



Bill number is auto generated and not changeable. After clicking submit button again index page will reappear to choose items (see above for index page). If you choose to issue “Samosha” the output will be as follow:-



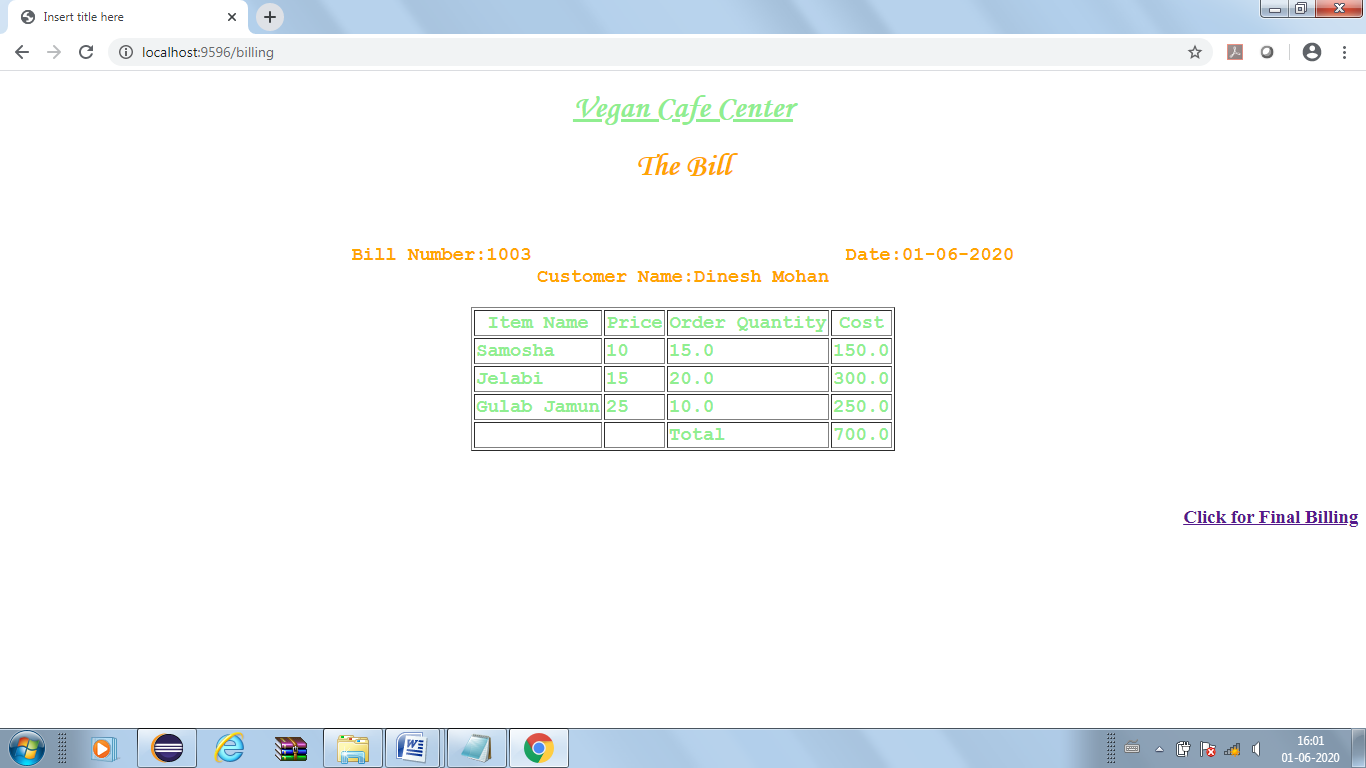
After clicking submit ,see the stock of Samosha is now less by 15 then the previous one.



In this manner for same customer, 20 Jelabi and 10 Gulab Jamuns are issued. The stock of them will also show the effect as follows:-



If you click *“Billing Process”* the bill will be generated as follows:-



Click for Final Billing will send the data into database. The index page will reappear.

Now if you click “Supply” option of Samosha then supply for Samosha will appear



Again index page appears as below:



If any item’s order quantity is higher than stock then an error page will appear. If item order starts, bypassing *new order* option then another error page will be displayed as there is no bill number generated.

**Create table *snack\_item* in Database.** The table has the following fields:-

item\_number(PK), item\_name, price, stock. PK->Primary Key

**Create table *snack\_order* in Database with the following fields**:-

bill\_number(PK),customer\_name,order\_date, total\_payment. PK->Primary Key

**Create table snack\_order\_list in Database with the following fields**:-

bill\_number,item\_number,item\_name,price, ordered\_quantity,amount\_payable.

(bill\_number,item\_number) composite primary key.

**Create application.properties file** to connect database from Java and server port settings

*Entry will be likely as follows:*

spring.datasource.url=jdbc:mysql://localhost:3306/<database-name>

or

spring.datasource.url=jdbc:oracle:thin:@localhost:1521:<database\_name>

spring.datasource.username=<user-name>

spring.datasource.password=<password-value>

**Create all model classes and attributes based on database tables.**

**Create getter and setter methods.**

**Create JpaRepository interfaces based on each table.**

**Create required Service classes to for database handling and business logic.**

**Create class SnackController** as Controller class

**Create separate folder jsp** for jsp files.