Invoice Web Application

Mohammed Salahadin

Abdulrahman Tawfiq

Shakar Abdulrahman

Introduction

We have created a web application to get a printed invoice as a hard copy and store it in the database as a softcopy for the customer instead of writing it down manually. This software will be controlled and configured by the shell script which makes it a lot easier on the user to install and configure it by giving the user few options to choose from. After the user has the shell script on his linux system, he will be able to run it in root privileges in order to install the software on his machine. When the user runs the shell script, he will see the instructions and options to use the system.

Background Research

According to our research we have found that shops owners still use the traditional way of selling items by using the regular invoice. There are many other solutions for this problem out there but those solutions are too comprehensive that doesn't satisfy the current requirements. That's why we want to implement it ourselves to small software that can accomplish the specific task with many advantages and without any overhead, as well as the program should be implementable and easy to use, that's why we decided to implement it to be easier than the others in terms of usability.

Methodology and Approach

A- We are using the Mysql as a database for this software

MySQL DBMS
The most important features of MySQL database systems are speed and reliability, which explains why they are frequently used by developers, administrators, and users around the world. Here are some reasons why I chose this system to be used in the DVD Movie Rental Shop application.
1- Security in MySQL:
Security is a strong point in MySQL, as it comes with a complex access control system and an authorization system to prevent unauthorized users from accessing the database. This system is implemented in the form of five layers of powers hierarchically, as MySQL administrators can protect access to sensitive data. It can also allow users to perform operations on specific databases or only
specific fields.

2- Ease of use MySQL:
Ease of use is an important point that MySQL has focused on. The MySQL development team has taken on the task of facilitating the use, management and improvement of MySQL performance. The main interface of MySQL is a simple inline interface, which consists of two graphical interfaces which are
MySQL Control Center and MySQL Administrator, which were developed by MySQL AB to use and manage MySQL.
3- Support various programming languages:
MySQL provides a programming interface for various programming languages to enable you to write database applications in the language of your choice. It supports PHP, Java, c ++, Perl, Python, Tcl and others to give developers maximum freedom in designing applications that rely on MySQL.

B- Using different programming languages
Php is a high-level versatile programming language. A high-level programming language where it is characterized by its ease and use of understandable English terms and hides a lot of details to deal with computer hardware unlike the low-level languages. Accompanied with other necessary programming languages like javascript and web technologies like html and css and other libraries like bootstrap and jquery.

C- Shell Script
We have used a linux shell script file which is the main file for running the system to install, control, eidt, backup or run the system.

The Advantages of using this software are:

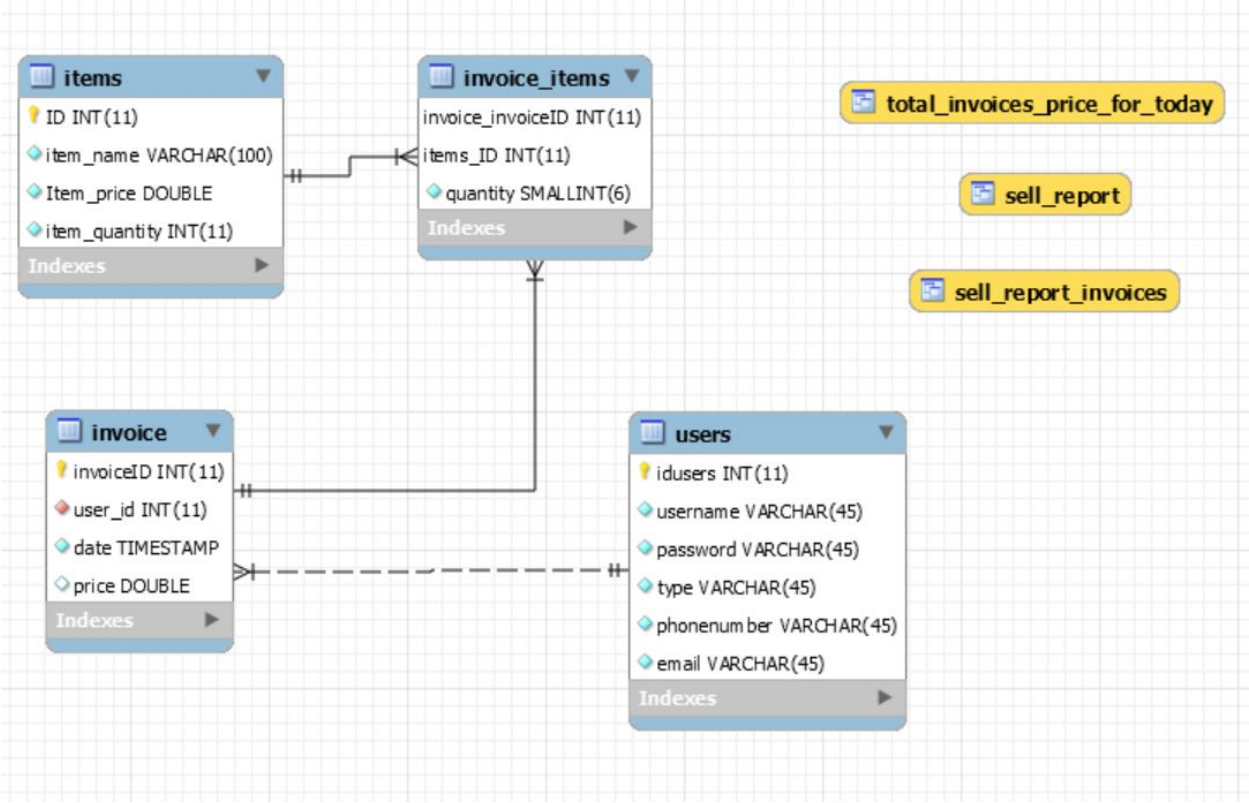
1. Use this software to get weekly or monthly reports.
2. Get information about how many items left in the store.
3. Get information about the expired or close to expire items in order to take an action
4. Get information about the sold items and who sold them including the date and time of selling them.

Implementation

Database Design

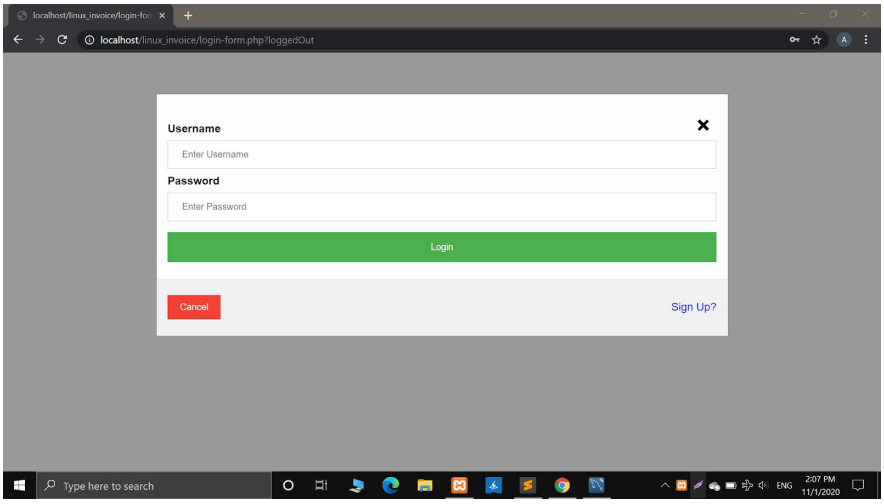
Creating necessary tables for the project

Designing the home page front-End

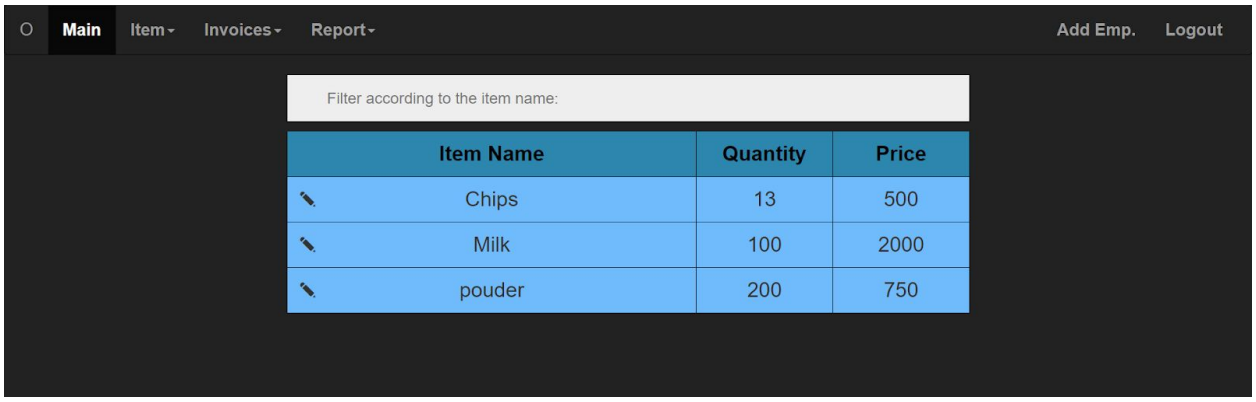


The Website

Screenshot of the login interface:



Items Table

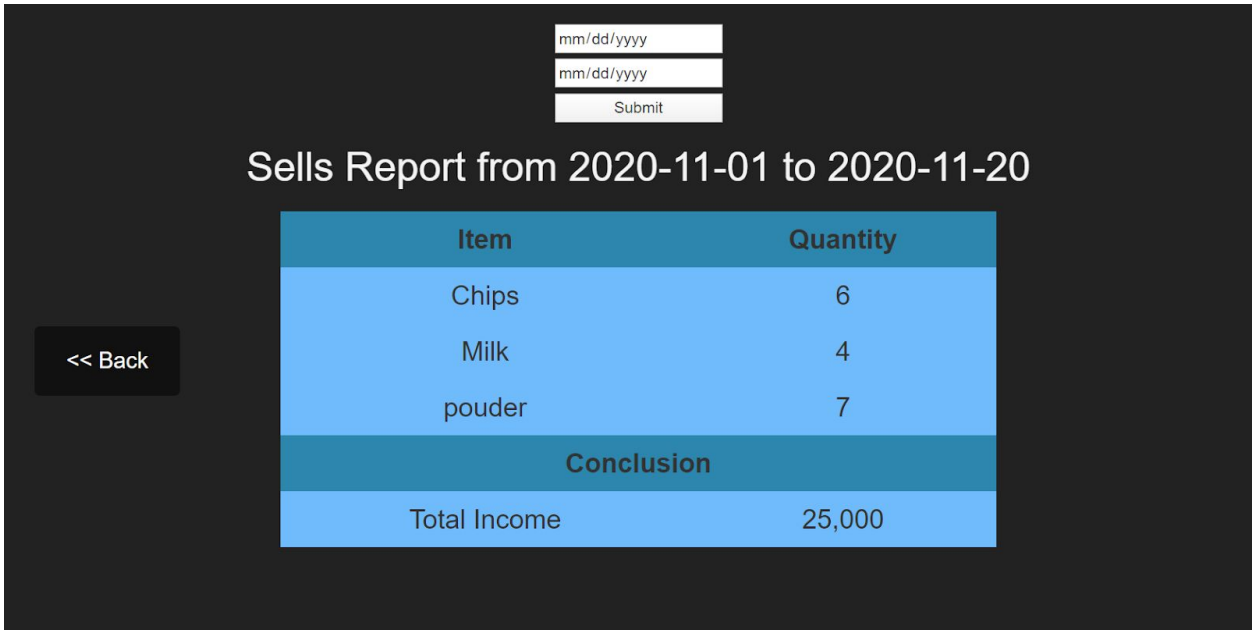


Daily Report

Report of day: 2020/12/11

Sold Section	
Item	**Quantity Sold**
chips	4
Invoice Section	
Invoice No.	**Invoice Price**
32	2,000
33	2,000
Conclusion	
Today sales price	4,000

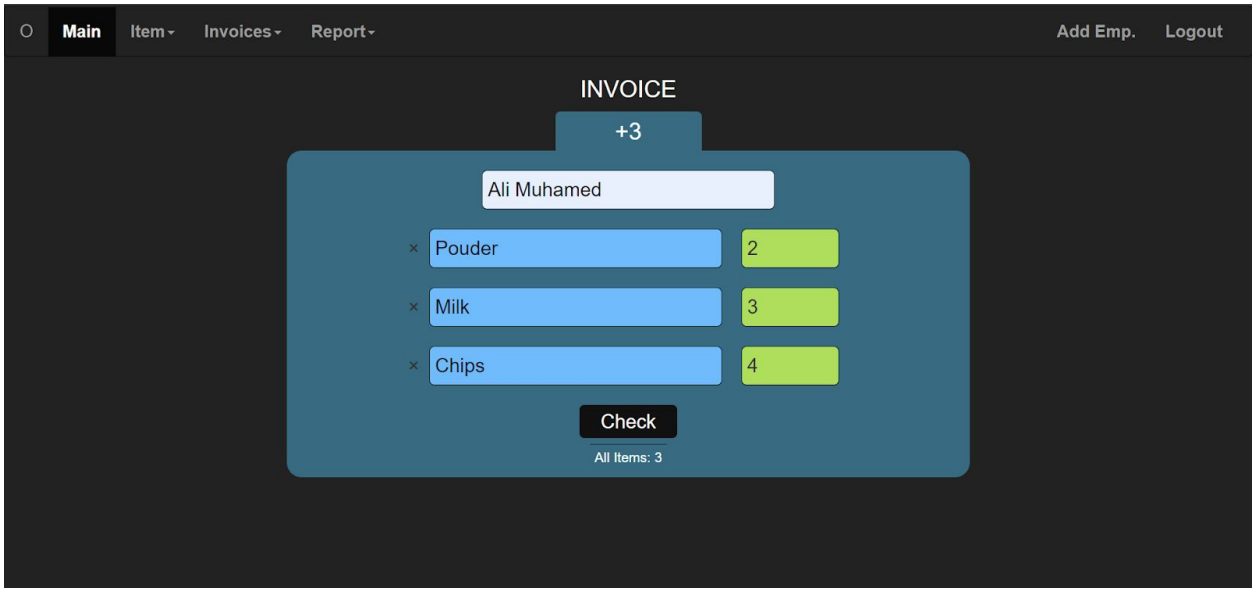
Report According to the Date

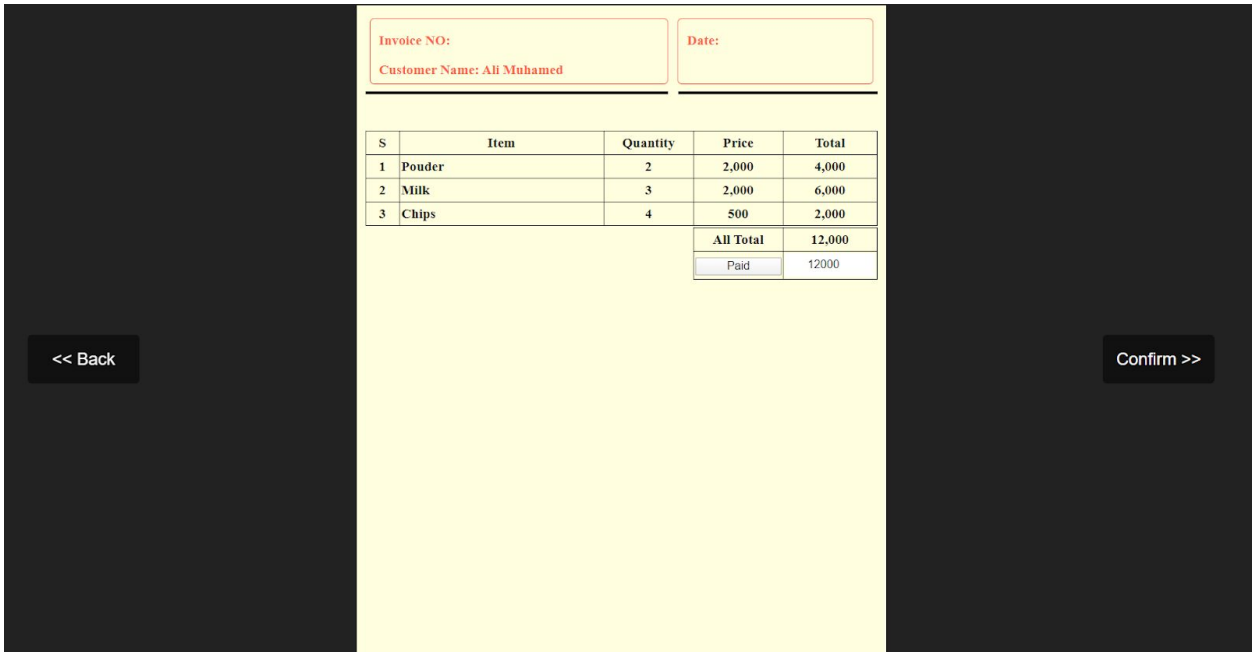


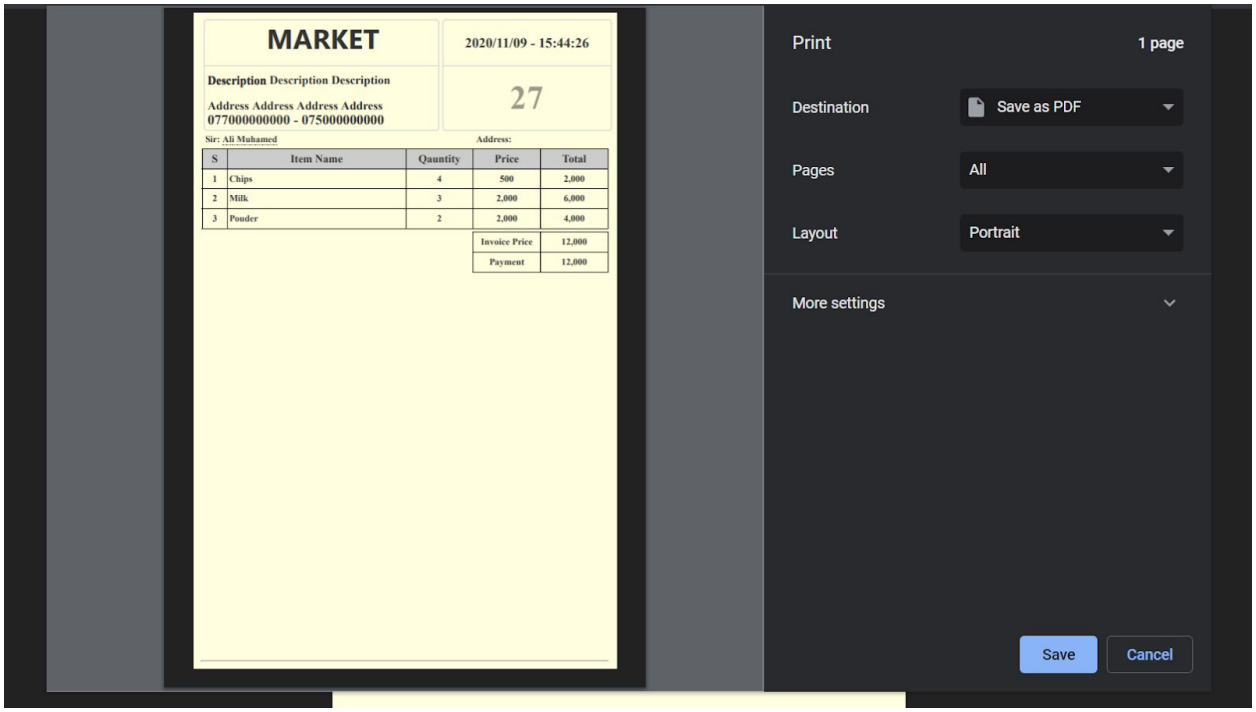
Item	Quantity
Chips	6
Milk	4
pouder	7
Conclusion	
Total Income	25,000

Below image is for inserting the items in the form.

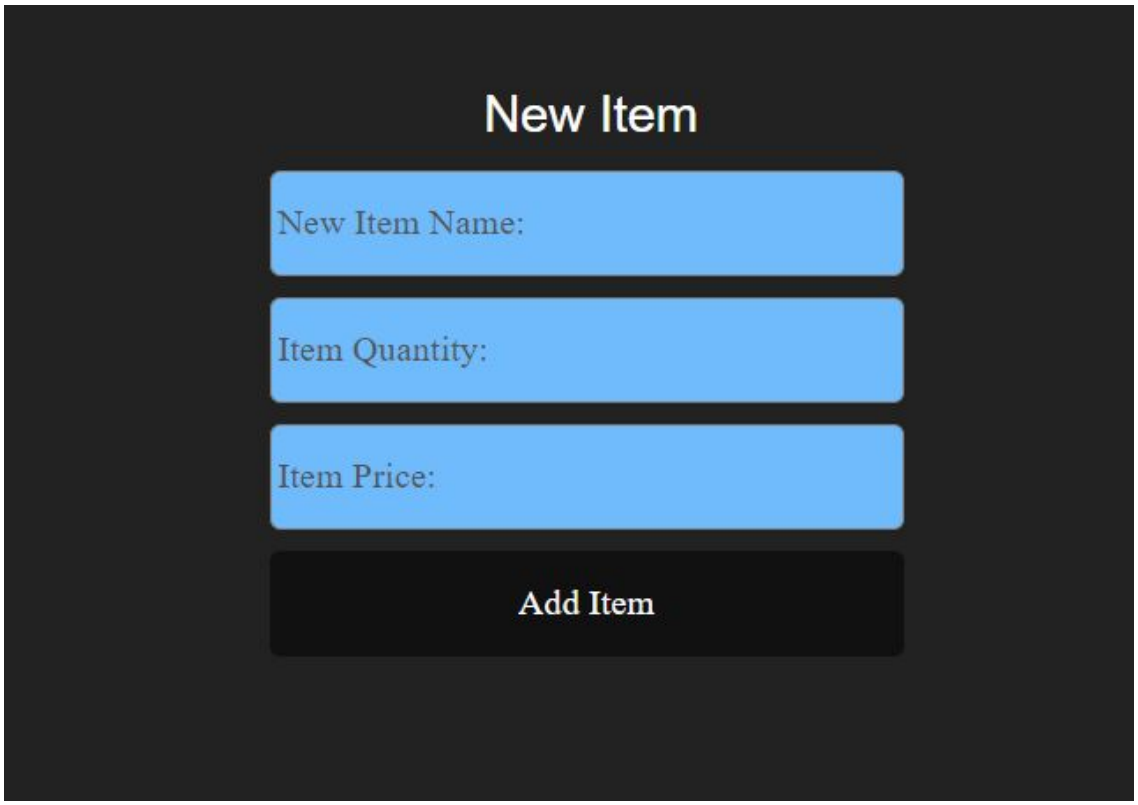
The employee will fill the form



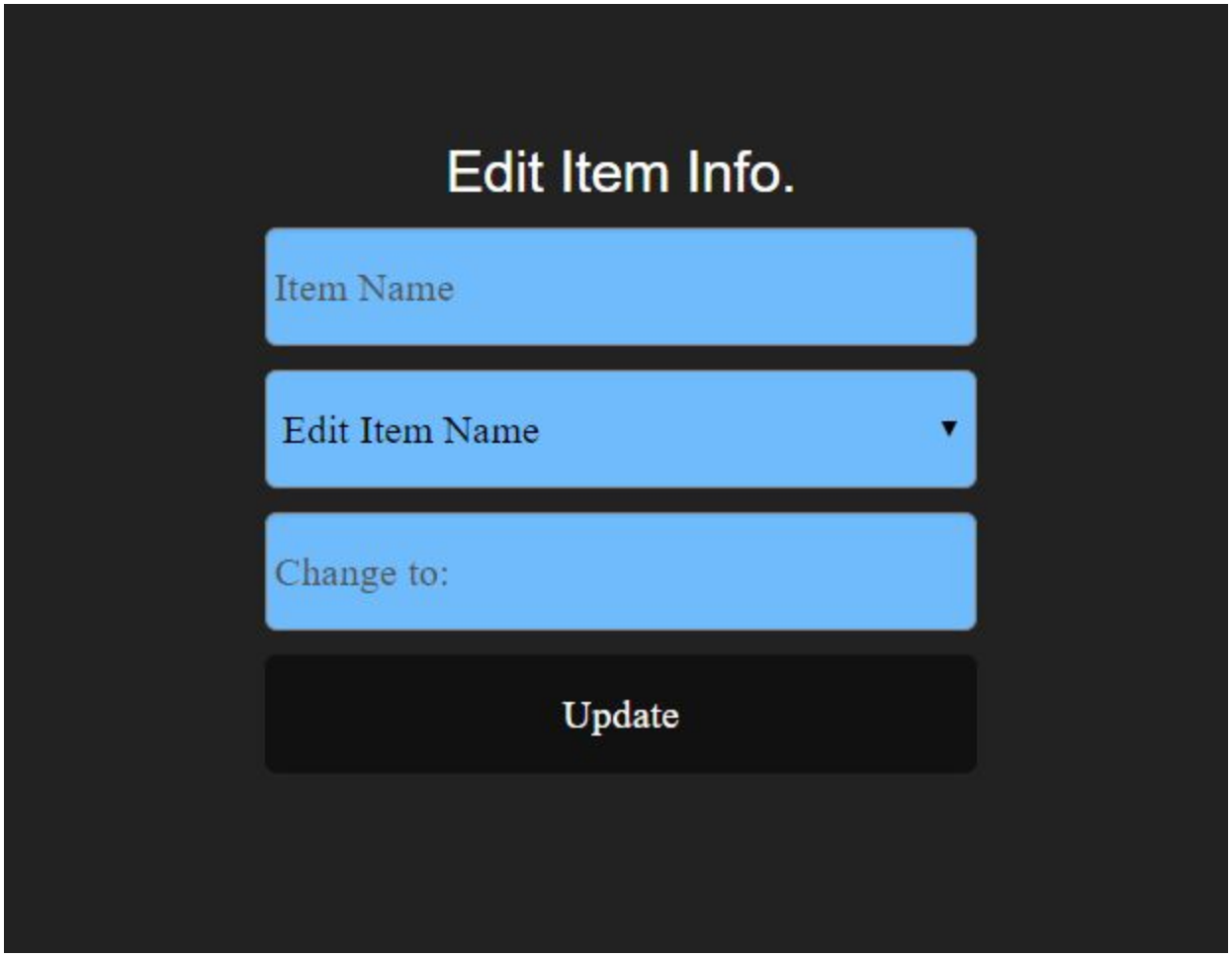




After Entering the items and clicking on check this page will appear



Here is the page for inserting new items to the database

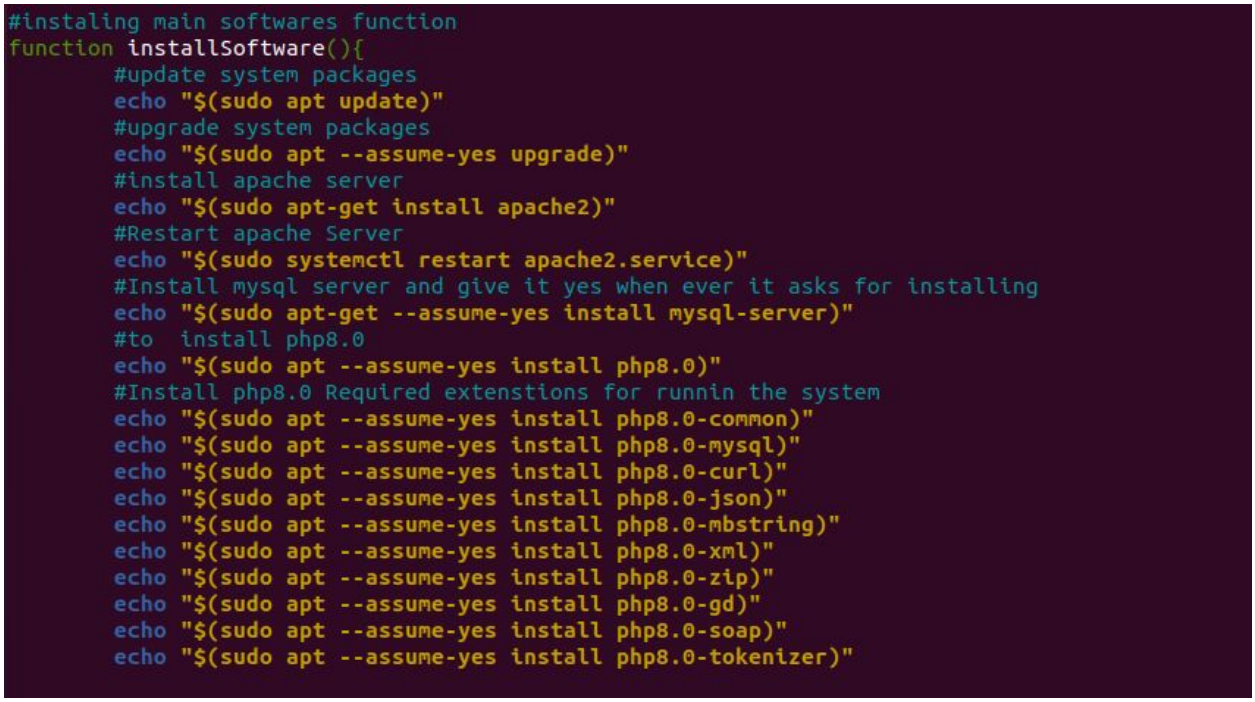


To update existing items in the database

Main Criterias and functionality of the project

- ### installSoftware

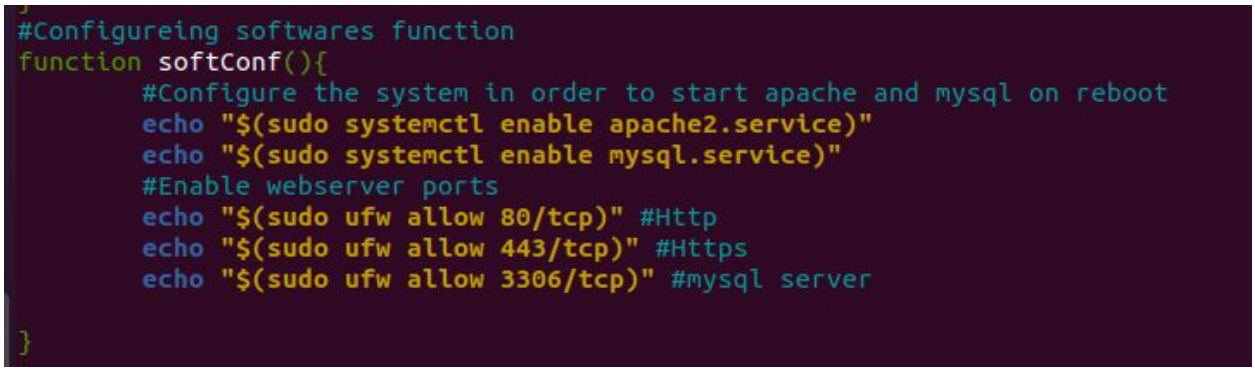
  will contain the codes for installing all the required software which are necessary for running the system:



```
#instaling main softwares function
function installSoftware(){
        #update system packages
        echo "$(sudo apt update)"
        #upgrade system packages
        echo "$(sudo apt --assume-yes upgrade)"
        #install apache server
        echo "$(sudo apt-get install apache2)"
        #Restart apache Server
        echo "$(sudo systemctl restart apache2.service)"
        #Install mysql server and give it yes when ever it asks for installing
        echo "$(sudo apt-get --assume-yes install mysql-server)"
        #to  install php8.0
        echo "$(sudo apt --assume-yes install php8.0)"
        #Install php8.0 Required extenstions for runnin the system
        echo "$(sudo apt --assume-yes install php8.0-common)"
        echo "$(sudo apt --assume-yes install php8.0-mysql)"
        echo "$(sudo apt --assume-yes install php8.0-curl)"
        echo "$(sudo apt --assume-yes install php8.0-json)"
        echo "$(sudo apt --assume-yes install php8.0-mbstring)"
        echo "$(sudo apt --assume-yes install php8.0-xml)"
        echo "$(sudo apt --assume-yes install php8.0-zip)"
        echo "$(sudo apt --assume-yes install php8.0-gd)"
        echo "$(sudo apt --assume-yes install php8.0-soap)"
        echo "$(sudo apt --assume-yes install php8.0-tokenizer)"
```

- ### SoftConf

  will contain all the codes which are responsible for configuring the installed software and making sure the service is always available even when the user restarts the system.

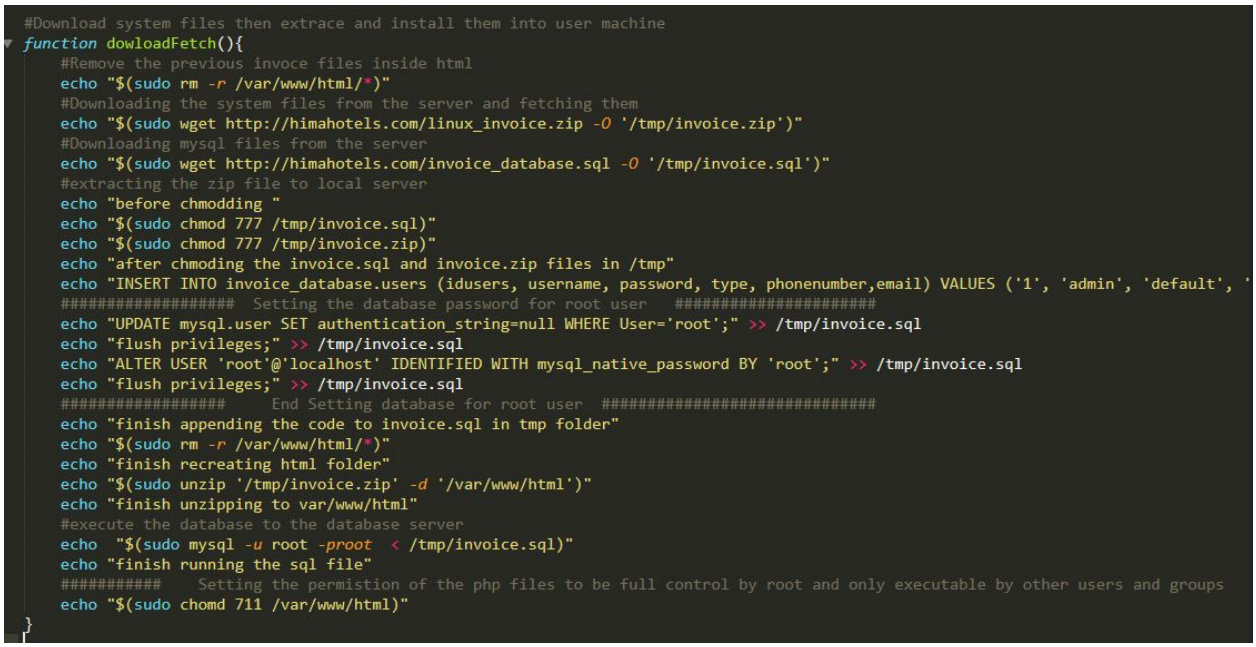


```
#Configureing softwares function
function softConf(){
        #Configure the system in order to start apache and mysql on reboot
        echo "$(sudo systemctl enable apache2.service)"
        echo "$(sudo systemctl enable mysql.service)"
        #Enable webserver ports
        echo "$(sudo ufw allow 80/tcp)" #Http
        echo "$(sudo ufw allow 443/tcp)" #Https
        echo "$(sudo ufw allow 3306/tcp)" #mysql server

}
```

- ## Download and fetch system files

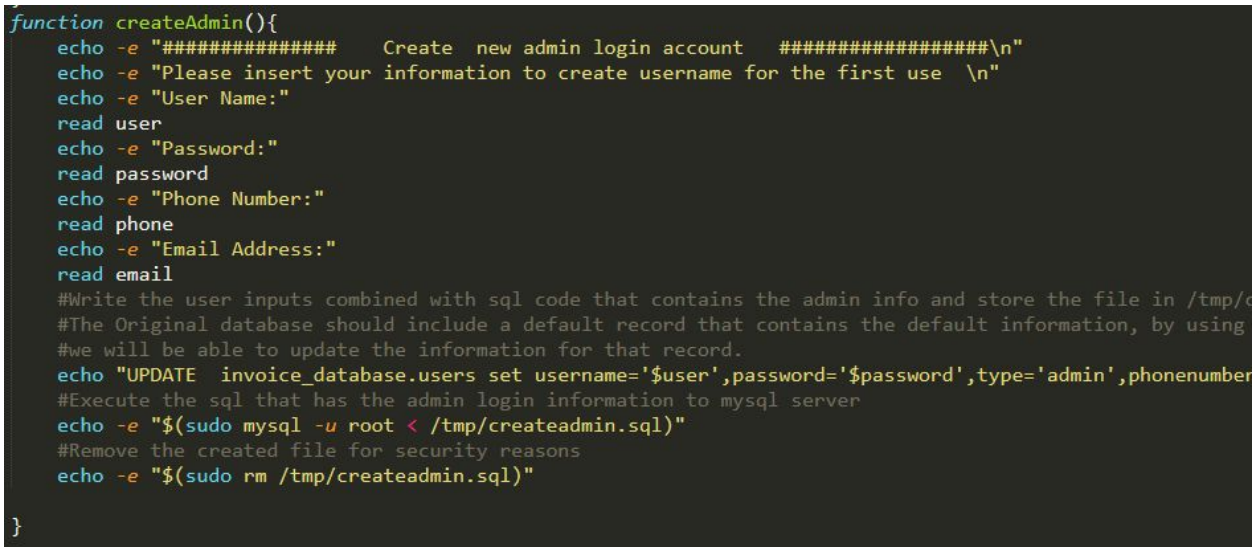
  This code is responsible for installing all softwares required for running the system.



```
#Download system files then extrace and install them into user machine
function dowloadFetch(){
    #Remove the previous invoce files inside html
    echo "$(sudo rm -r /var/www/html/*)"
    #Downloading the system files from the server and fetching them
    echo "$(sudo wget http://himahotels.com/linux_invoice.zip -O '/tmp/invoice.zip')"
    #Downloading mysql files from the server
    echo "$(sudo wget http://himahotels.com/invoice_database.sql -O '/tmp/invoice.sql')"
    #extracting the zip file to local server
    echo "before chmodding "
    echo "$(sudo chmod 777 /tmp/invoice.sql)"
    echo "$(sudo chmod 777 /tmp/invoice.zip)"
    echo "after chmoding the invoice.sql and invoice.zip files in /tmp"
    echo "INSERT INTO invoice_database.users (idusers, username, password, type, phonenumber,email) VALUES ('1', 'admin', 'default', '
    ##################  Setting the database password for root user   ####################
    echo "UPDATE mysql.user SET authentication_string=null WHERE User='root';" >> /tmp/invoice.sql
    echo "flush privileges;" >> /tmp/invoice.sql
    echo "ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'root';" >> /tmp/invoice.sql
    echo "flush privileges;" >> /tmp/invoice.sql
    #################     End Setting database for root user  ############################
    echo "finish appending the code to invoice.sql in tmp folder"
    echo "$(sudo rm -r /var/www/html/*)"
    echo "finish recreating html folder"
    echo "$(sudo unzip '/tmp/invoice.zip' -d '/var/www/html')"
    echo "finish unzipping to var/www/html"
    #execute the database to the database server
    echo  "$(sudo mysql -u root -proot  < /tmp/invoice.sql)"
    echo "finish running the sql file"
    ###########    Setting the permistion of the php files to be full control by root and only executable by other users and groups
    echo "$(sudo chomd 711 /var/www/html)"
}
```

- ## CreateAdmin

  Contains all the commands required for creating a new admin account to manage the system, it depends on the user input for setting the username and the password for the first time, this account will be stored in the database in order to use it later for login to the system.



```
function createAdmin(){
    echo -e "###############    Create  new admin login account   ##################\n"
    echo -e "Please insert your information to create username for the first use  \n"
    echo -e "User Name:"
    read user
    echo -e "Password:"
    read password
    echo -e "Phone Number:"
    read phone
    echo -e "Email Address:"
    read email
    #Write the user inputs combined with sql code that contains the admin info and store the file in /tmp/c
    #The Original database should include a default record that contains the default information, by using
    #we will be able to update the information for that record.
    echo "UPDATE  invoice_database.users set username='$user',password='$password',type='admin',phonenumber
    #Execute the sql that has the admin login information to mysql server
    echo -e "$(sudo mysql -u root < /tmp/createadmin.sql)"
    #Remove the created file for security reasons
    echo -e "$(sudo rm /tmp/createadmin.sql)"

}
```

- Install all functions

  It’s the main function which users call when they want to install the software on their machine, this function is responsible for calling the above functions.

```
#For installing Invoice software
function installall(){
        ###  Installing required software and configure it for running the system
        installSoftware
        #software configuration
        softConf
        ### For Downloading and fetching the system
        dowloadFetch    #Create the first admin account
        createAdmin
        #running the system
        run
}
```

- System configuration

  This function is responsible for configuring the invoice system, like changing the host or adding new employee

```
function configuration(){
        echo "Welcome to configure your invoice system"
        echo "########################################"
        echo "1 To update you localhost Domain name"
        echo "2 To insert new Employee Account"
        read input2
        if [[ $input2 == 1 ]];then
                localhost_change
        elif [[ $input2 == 2 ]];then
                addEmployee
        else
                echo "You have chosen a wrong option"
        fi

}
```

- ## Localhost Change

  To change the hostname to another

```
function localhost_change(){
        echo -e "##################  Change the site domain name    ###################"
        echo -e "Please Enter the new domain name:"
        read admin
        #give premision to modefy host and hostname files
        echo "$(sudo chmod 777 /etc/hostname)"
        echo "$(sudo chmod 777 /etc/host)"
        #change the localhost 127.0.0.1 to the admin name
        echo "$(sudo echo "$admin" > /etc/hostname)"
        echo "$(sudo echo "127.0.0.1 $admin" > /etc/host)"
        echo "$(sudo hostname "$admin")"
        echo "$(sudo service apache2 restart)"
echo "Your domain name is now: http://$admin"
}
```

- ## Add Employee

  Through the shell script admin will be able to add a new user. Once the new user is added, he will be a new linux user as well, and will be able to use the mail service.

```
function addEmployee(){
        echo "User Name:"
        read userName
        echo "Password:"
        read password
        echo "Phone Number:"
        read phone
        echo "Email:"
        read email

        #Create an sql file of contains
        #we will be able to update the information for that record.
        echo "INSERT INTO invoice_database.users (username, password, type, phonenumber, email) VALUES ('$userName', '$password', 'user', '$phone', '$email');" > /tmp/adduser.sql
        #Execute the sql that has the admin login information to mysql server
        echo -e "$(sudo mysql -u root -proot < /tmp/adduser.sql)"
        #Remove the created file for security reasons
        echo -e "$(sudo rm /tmp/adduser.sql)"
        echo "User $userName Have been added to the systen"
        #Create New linux user using this username and password
        echo "$(sudo useradd -m $userName)"
        echo "$(sudo echo "$userName:$password" | sudo chpasswd)"
        echo "$(sudo usermod -aG mail $userName)"
        echo "User $userName Have been added to the mail group"
```

- ## Daily Report

  Getting daily reports.

```
function daily_report(){
        #this command will execut the command
        # should specify the user name and password for login to the database by using -u for the username and -p for the password
        #-h is used to connect to the host that is running the database
        #-D to select the database that we are going to work with
        # -e The most  importatn paramater that sends the query to the database to execute it.
        echo "####### Getting daily report #########"
        echo "$(mysql -u root -proot -h 127.0.0.1 -D invoice_database -e 'SELECT * FROM invoice_database.sell_report;SELECT * FROM invoice_database.sell_report_invoices;SELECT * FRO
}
```

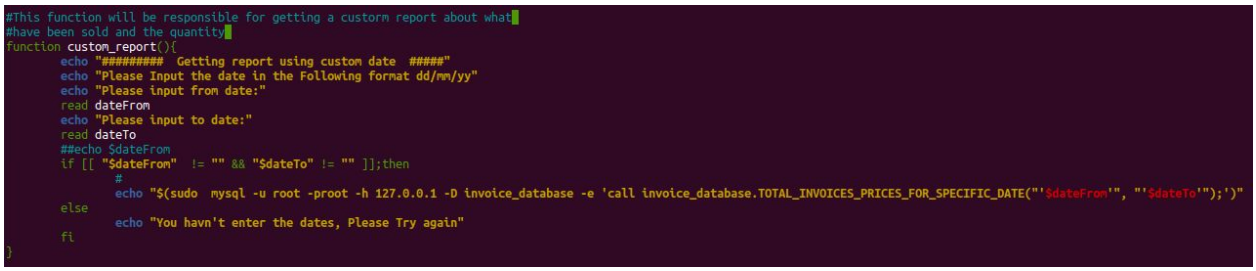
```
+----------+-------+---------+
| goods_ID | Name  | No_of_k |
+----------+-------+---------+
|       16 | chips |       4 |
+----------+-------+---------+
+-----------+-------+---------------------+
| invoiceID | price | date                |
+-----------+-------+---------------------+
|        32 |  2000 | 2020-12-11 17:13:01 |
|        33 |  2000 | 2020-12-11 17:13:39 |
+-----------+-------+---------------------+
+-----------------+
| totInvoicePrice |
+-----------------+
|            4000 |
```

Report of day: 2020/12/11

Sold Section	
Item	**Quantity Sold**
chips	4
Invoice Section	
Invoice No.	**Invoice Price**
32	2,000
33	2,000
Conclusion	
Today sales price	4,000

- Custom Report

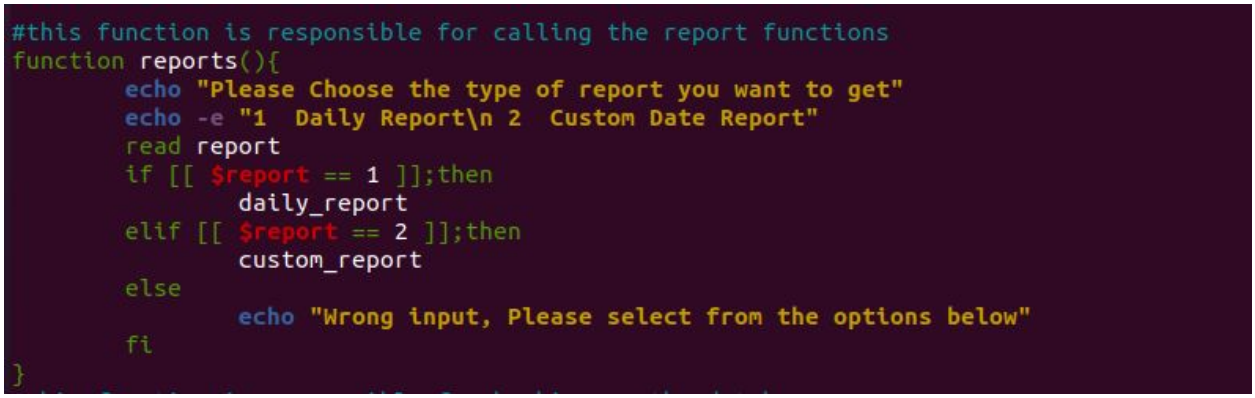
  Getting a report according to the user input.



```
#This function will be responsible for getting a custorm report about what
#have been sold and the quantity
function custom_report(){
        echo "########  Getting report using custom date  #####"
        echo "Please Input the date in the Following format dd/mm/yy"
        echo "Please input from date:"
        read dateFrom
        echo "Please input to date:"
        read dateTo
        ##echo $dateFrom
        if [[ "$dateFrom"  != "" && "$dateTo" != "" ]];then
                #
                echo "$(sudo  mysql -u root -proot -h 127.0.0.1 -D invoice_database -e 'call invoice_database.TOTAL_INVOICES_PRICES_FOR_SPECIFIC_DATE("'$dateFrom'", "'$dateTo'");')"
        else
                echo "You havn't enter the dates, Please Try again"
        fi
}
```

- Reports

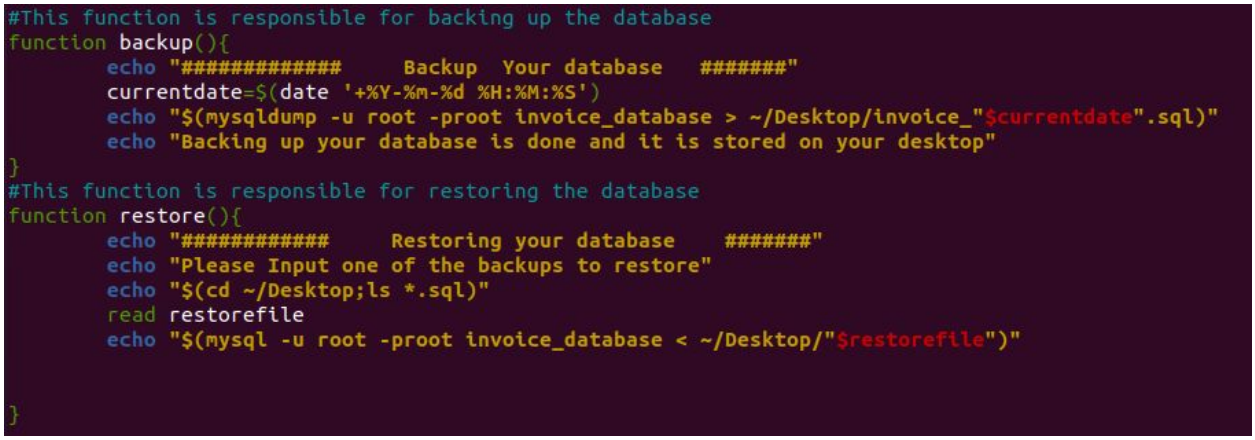
  Is responsible for calling the rest of the methods for daily and custom reports.



```
#this function is responsible for calling the report functions
function reports(){
        echo "Please Choose the type of report you want to get"
        echo -e "1  Daily Report\n 2  Custom Date Report"
        read report
        if [[ $report == 1 ]];then
                daily_report
        elif [[ $report == 2 ]];then
                custom_report
        else
                echo "Wrong input, Please select from the options below"
        fi
}
```

- Backup and restore database

  These two functions are responsible for backing up the database and restore them



```
#This function is responsible for backing up the database
function backup(){
        echo "##############     Backup  Your database   #######"
        currentdate=$(date '+%Y-%m-%d %H:%M:%S')
        echo "$(mysqldump -u root -proot invoice_database > ~/Desktop/invoice_"$currentdate".sql)"
        echo "Backing up your database is done and it is stored on your desktop"
}
#This function is responsible for restoring the database
function restore(){
        echo "#############     Restoring your database    #######"
        echo "Please Input one of the backups to restore"
        echo "$(cd ~/Desktop;ls *.sql)"
        read restorefile
        echo "$(mysql -u root -proot invoice_database < ~/Desktop/"$restorefile")"


}
```

- Removing the system

  User will be able to remove the system from the machine easily using only one command:

```
function removeSystem(){
        echo "$(mysql -u root -proot -h 127.0.0.1 -D invoice_database -e 'drop database invoice_database;')"
        echo "$(sudo rm -r /var/www/html ; sudo mkdir /var/www/html)"
        echo "${Green} System have been removed ${NC}"
}
```

- Outputs colors

  Users will see outputs of the program with colors, depending on the stage and type of the messages.

```
#Defingin colors for using them in the program
Green='\033[1;32m'
Red='\033[1;31m'
Cyan='\033[1;36m'
Yellow='\033[1;33m'
NC='\033[0m' #End of the color
```

Results

We have created a shell script that installs and configures all the required software for running the invoice system from linux terminal. The admin will be able to install, configure, backup, restore and run the system easily by following the instruction guide which will be shown in the terminal.

After installation of the system, the user will be automatically redirected from the shell script to the local website that runs the system in order to start working on it, through the shell script user also will be able to show reports of the shop.

Users will be able to add new users as employees to the system, they will be added to the database of the system at the same time they will have a new linux account with the same username and password, also they will be added to the mail group list, to send, receive emails internally.

Conclusion

In conclusion, using linux terminal users will be able to install invoice systems easily through install option, once the user selects the installation option all the necessary softwares which is responsible for running the system will be installed and configured automatically. During the installation process users will be asked to configure the login to the system by providing a user name and password, by using the terminal and after the installations of the system the shell script will be able to detect if the system have been installed successfully, through the options user will be able to control the hole system via choosing from the options.

References

1. https://www.linux.org/
2. https://ubuntu.com/
3. https://stackoverflow.com/
4. https://cheatography.com
5. https://maker.pro/
6. https://www.guru99.com/