CHAPTER1

INTRODUCTION

In this, we use the functions like insert item, delete item, display item and display the item and purchase the item what we insert into the cart. In the insert function we can insert the items with item id, Item name, item cost. For displaying we use the display function to display the available items in the stock. If the product is sale the function show it paid. For deleting the item, we use delete function to delete the particular item. If we want to buy the particular product first, we have to insert the item in to the cart after that we should buy the product.

This project relates the python concept in the eco live items. The problem statements of this project are as follows:

- In this project we input many items like fertilizers, pesticides, insecticides etc. The items are selected or add by the customer after adding the item. The customer can purchase the item.
- A main class is defined to input the items and display the items in the class only. This project we use trees and database for storing the data and search the stored data items.
- Here we use python features like classes as mentioned above, trees and data base to implement this project.
- A class is a user defined data-type which has data base for storing the input items of the eco live.
- Now a day, all are utilizing eco lives for buying the products or items. The eco live are placed in every street and every area.
- In the eco live, many items are available to purchase the products what customers want. In the eco live, items are input by the admin person

1.1 PROBLEM DEFINATION

This project relates the python concept in the supermarket items. The problem statements of this project are as follows:

In this project we input many items like fertilizers, pesticides, insecticides etc. The items are selected or add by the customer after adding the item. The customer can purchase the item.

A main class is defined to input the items and display the items in the class only. This project we use trees and database for storing the data and search the stored data items. Here we use python features like classes as mentioned above, trees and data base to implement this project...

A class is a user defined data-type which has data base for storing the input items of the eco live.

Nowadays, all are utilizing eco live for buying the products or items. The eco live are placed in every street and every area.

In the eco lives, many items are available to purchase the products what customers want. In the eco live, items are input by the admin person.

1.2 OBJECTIVES

In this python code, we use classes, trees and database to store the data what we use in this code. In this python code, we use the functions like adding data, deleting data and displaying the data and the product is sale or not. In the adding function admin can insert the items and delete the particular item and displayed the items what in the store. After that the product is sale or not. If the product is sale then the item is displayed as not sale.

- Build the class with the instance functions or any other methods we can load the class.

- Functions
- Call the methods or any function inside the program.
- In this code, we database to store the data what we insert into the python code.

1.3 EXPECTED OUTCOMES

The outcomes are expected as below-

1. The customer can select the items what he/she want. The items are searched by the product name. And the item will be adding on the eco live kart.
2. After adding the items from the ECO LIVE store, they must be buying the product. By using the payment method. There is no modification can do when the items bought. Python concepts such as class, tress and data base are achieved in this project.

CHAPTER 2

REQUIREMENT SPECIFICATIONS

2.1 HARDWAREREQUIREMENTS

Processor : Any Processor above 500 MHz

RAM : 512Mb

Hard Disk : 10 GB

Input device : Standard Keyboard and Mouse

Output device : VGA and High Resolution Monitor

2.2 SOFTWARE REQUIREMENTS

Operating system : Windows XP

Front End : ASP.Net 2.0

Server : Internet Information Services

Database Connectivity: ODBC Sources (with SQL Server)

CHAPTER 3

PYTHON FUNDAMENTALS

What is Python?

Python is a scripting programming language which is mainly known for both its simplicity and wide breadth of applications.

For this reason it is considered one of the best languages for beginners. It is used for everything from Web Development to Scientific Computing (and SO much more), Python is referred to be as a "general purpose" language by the greater programming community.

Many Python programmers (aka "Pythoness's") love this language because it maintains a certain philosophy of best practices, described in Tim Peter's famous "Zen of Python". There is a large Python community both off and online that is welcoming and supportive of beginners, and you can find a plethora of additional materials in the resources section of this guide.

Easy to Learn and Use

- **Easy to Code:**

  Python is easy and quick to learn and use. It is programmer-friendly and high level programming language. Python is very efficient to code. Compared to other programming languages like C++ and Java, it is easier to code in Python. One can learn python syntax in just a few hours. Thus, it is developer-friendly.

- **Easy to read:**

  Being a high-level language, Python code is just like English. Looking at it, one can tell what the code is intended to do. Further, since it is dynamically-typed, it enforces indentation. This aids better readability.

3.1 Expressive Language:

Python language is more expressive which means that it is more comprehendible and readable. Suppose there are two languages A and B, and all programs that can be developed in A can be made in B using local transformations. Whereas, there are some programs that can be developed in B, but not in A, using local transformations. Then, B is considered more expressive than A. Python provides a myriad of constructs that help developers focus on the solution rather than too much on the syntax.

The Python fundamental consists of basic building blocks of Python programming language. And it is basically divided into the following categories.

- Statements
- Indentations
- Comments
- Variables
- Constants
- Tokens

3.1.1. Statements:

They are logical instructions that interpreter can execute and read, it can also be both single and multi line.

The two categories of the Python Statements are:

- Expression Statements
- Assignment Statements

Expression Statement: By the help of expression statements, we can perform the operations like addition, subtraction, concentration and many more. In short, the statement has return value.

It is an expression that appears on the right side of the assignment, as a parameter to method call.

Assignment Statement: By the help of assignment statements we can create new variables, assign values and also change values.

Assignment statements are categorized into three:

- Value-Based Expressions on Right hand side
- Current Variables on Right hand side
- Operation on Right hand side

3.1.2. Indentation:

The programming languages python uses indentation to mark a block of the code. Most of the Programming languages provide indentation for better code formatting and doesn't enforce to have it. But mainly in Python it is mandatory.

That's why indentation is crucial in Python.

3.1.3. Comments:

Comments are basically nothing but tagged lines of in codes which increases the readability of the code and make the code self-explanatory.

There are two categories of Comments:

i. Single line Comments: '#' by the help of these we begin a single-line comment.

ii. Multi-line comments: '''…''' by the help of these we write multiline comments in python.

iii. Doctstring comments: The documentation string in Python gives programmers an easy way of adding qui k notes with every Python module, functions, class and method.

Multiline comments are using triple quotation in strings.

3.1.4. Variables:

In Python variable is a memory address that can change, when a memory address cannot change then it is known as constant. Variable is the name of the memory location where the data is stored. Once the variable is stored then space is allocated in memory. It also defines the variable using a combination of numbers, letters, and the underscore character.

3.1.5. Constants:

In Python constants is a type of variable that holds values, whose value cannot be changed. We rarely use constants in Python.

3.1.6. Token:

In Python tokens are the smallest unit of the program. Python contains the following tokens:

- Reserved words or keywords
- Identifiers
- Literals
- Operators

Reserved words: Reserved words are nothing but a set of special words, which are reserved by python and also have a specific meaning. Here, in Python we are not allowed to use keywords as variables. Reserved words are case sensitive in Python.

For example: False, if, none, import, True, in, and, def, return, elif, try, else, while, except, with, finally, yield, is, as, break, class, etc.

Identifiers: In Python identifiers are nothing but user-defined names to represent programmable entity like variables, functions, modules, classes. There are few rules that we need to follow while defining an identifier.

They are:

i. We can use a sequence of letters lowercase or uppercase. We can also mix up digits or an underscore while defining an identifier.

ii. We cannot use digit to begin an identifier name.

iii. We should not use reserved keywords to define an identifier.

iv. You are not allowed to use any other special characters other than underscore.

v. Even though python doc says that you can name an identifier with unlimited length.

Literals: Other built-in objects in python are literals. The Literals can be defined as data that is given in a variable or constant.

The following literals are in:

String Literals: String literals are a sequence of characters surrounded by quotes. Single, double or triple quotes can be used for a string.

Boolean literals: Boolean literal can have any of two values i.e true or false.

Numeric literals: They are immutable. Numeric literals can belong to three different numerical types Integer, Float, Complex.

Collection literals: The four types of collection literals are List literals, Tuple literals, Dict literals, and Set literals.

Special literals: Python basically contains one special character that is none.

Operators: In python operators are the symbols which perform the operation on some values. The following are the known operators in the Python.

- Arithmetic Operators
- Relational Operators
- Assignment Operators
- Logical Operators
- Membership Operators
- Identity Operators
- Bitwise Operators

3.2.TKINTER WIDGETS

LABEL WIDGET:

A label widget shows any text to the user. A text can be added programmatically or by default to able to display a text on the screen for the user. A label can be placed on the window or fame or a canvas.

BUTTON WIDGET:

A Button can be on and off. When a user clicks it, the button emits an event. Images can be displayed on buttons too. A button has a command and a function associated with it to perform an action when it is passed.

ENTRY WIDGET:

An Entry widget is used to accept a text input from the user. The obtained input text can be stored in a variable and used for several operations. The entry box can be placed on the window or frame and can accept values of several types such as StringVar(), IntVar() and DoubleVar().

SCALE WIDGET: HORIZONTAL

Scale widget is used to have a slider that goes from one value to another. The starting, ending and step values can be set. The current value of the slider can be accessed by its

Get method and the current value can be set by its set method.

LISTBOX WIDGET:

Listbox displays a list of items and lets the user choose from one set of options. The list can be of various types and the current value of the list box which is selected can be bound to an event and is represented by <<ListSelect>>.

3.3. DATABASE/ALGORITHM FUNDAMENTALS:

It introduces database concepts, relational database, tables and data types, manipulation and data selection, views, stored procedures, backup and restores normalization, constraints, indexes, security, and functions.

Tables: It is most basic building of a database. It's the place where we will put our idea, and define their data type, and also their relationship with the other tables. It consists of rows and columns.

Columns consist of three types: - Simple, Composite, Multi-valued

Primary key: If we have a long list of rows, it's very important to have something that uniquely

Identify each row that is called primary key.

And we are going to use primary key to connect between the tables, and to form a relationship.

The kinds of relationships are:

• One-to-Many Relationship

• Many-to-Many Relationship

SQL: The structured query language is the defect language used for the management and manipulation of data in relational database. The Sql can be used by query, insert, update, and modify data.

Select: The selecting statements of the retrieves zero or more rows from the one or more database tables.

1. SQL join combines the records from two or more tables in a relational database.

2. CROSS join will produce the row which is the combination of each row from the first table with each row from the second table.

3. AN INTERSECTION combines the results of two queries and it returns only rows that appear in both result sets.

4. A UNION enables to combine the results of two SQL queries into a single table of all matching row.

Manipulate data:

1. Insert data: - The statement INSERT adds one or more records for any single table in a relational database.

2. Update data: - The statement UPDATE changes the data of one or more records in a table.

3. Delete data: - DELETE statements remove one or more records from the table.

Views:

Create views: Create views are the result of set of a stored query on the data, where the database users can query just as they would persistent database collection object.

Stored Procedures: It is a subroutine available to applications that access a relational database management system.

Functions:

A user defined functions are provided by user and aggregate function is a function where the multiple values of rows are grouped together as an input on a certain criterion to form single value of more significant meaning.

Normalization:

Normalization is the process of an organizing of the columns and tables of a relational database to minimize data redundancy.

The types are First normal form (1NF), Second normal form (2NF), Third normal form (3NF), Fourth normal form (4NF).

Constraints:

We have to choose appropriate primary keys, select appropriate data type, select appropriate fields for composite keys, understand the relationship between foreign and primary key.

Indexes:

They are basically used to quickly to locate data without having to search each and every row in database table every time a database table is accessed.

Security:

The Database security is one of the use of a board range of an information security controls to the store functions, protect of databases, data servers, database systems and against compromises of their confidentiality, integrity, and availability.

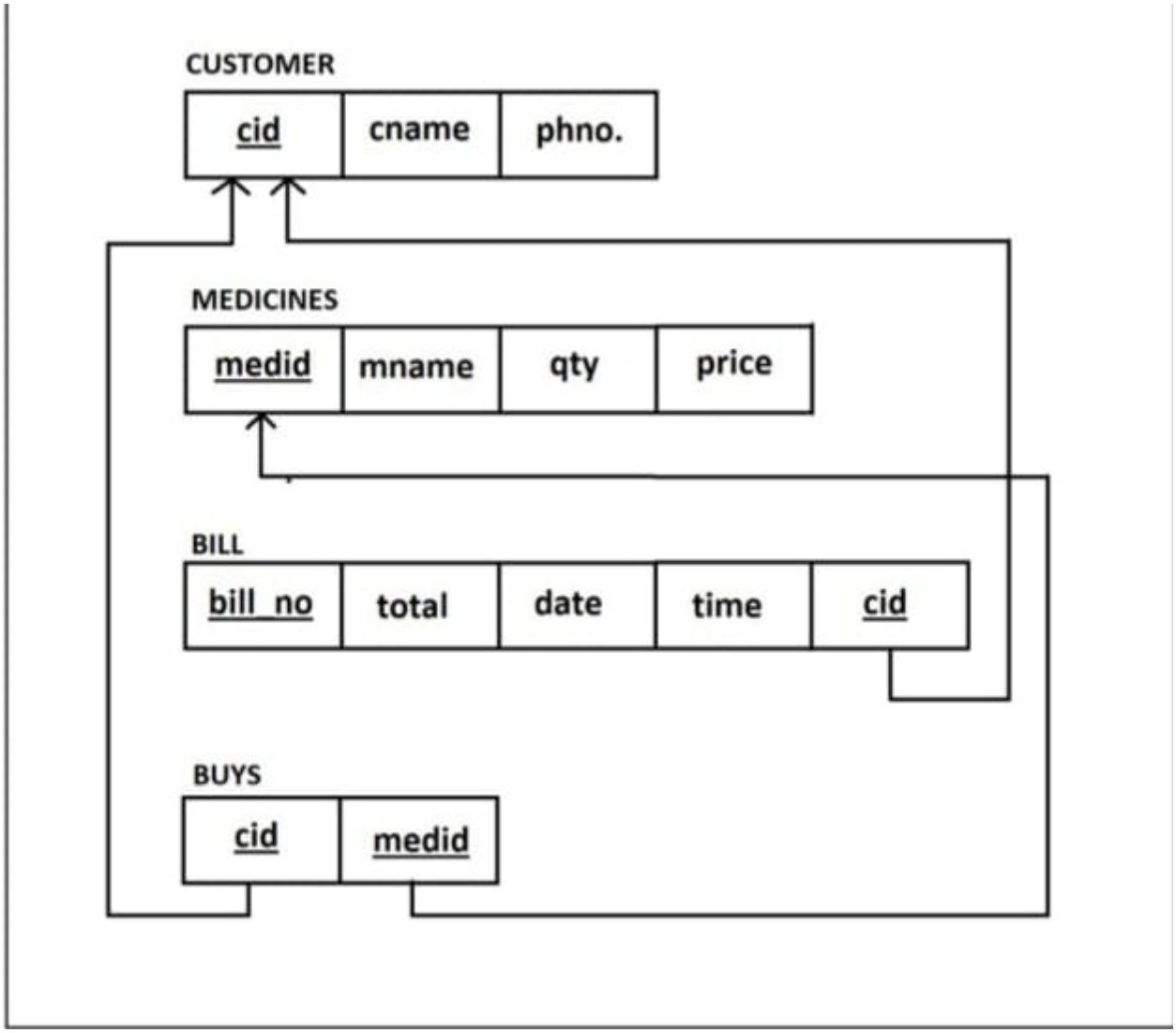
Backup and restore:

The process of backups refers to the copying the computer data and archiving of computer data so it may be used to restore the original after a data loss event.

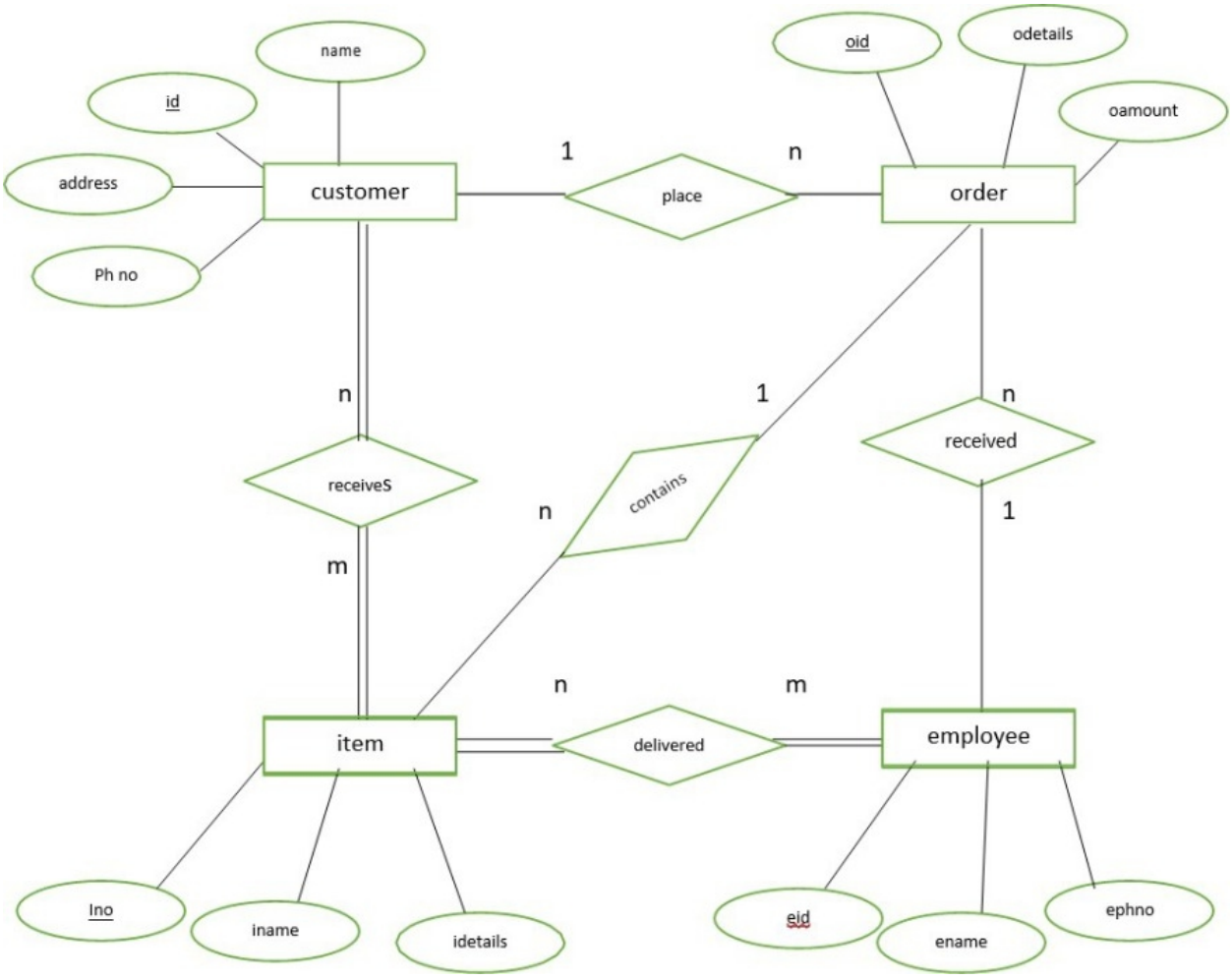
CHAPTER 4

ALGORITHM

4.1.Schema diagram



4.2.ER diagram



CHAPTER 5

IMPLEMENTATION

Implementation of Function

- A function is a block of code which only runs when it is called for.
- We can pass data, known as parameters, into a function when required.
- "def" is the keyword used to define a function.
- A function can return the value.

SYNTAX FOR CREATING A FUNCTION

```
Def my_function       //creating a function
    print ("Hello, I am block of function code")          //block  of code
my_function()          //calling function.
```

Implementation of Tkinter

- Tkinter is the standard GUI application library for Python.
- Fast and easy way to create GUI applications when they are required.
- It is a powerful object-oriented interface to the Tk GUI toolkit.

  Import the Tkinter module.

SYNTAX FOR CREATING A TKINTER

```
import Tkinter
    top = Tkinter.Tk()
   # Code to add widgets will go here...
top.mainloop()
```

Implementation of Framework widget

- ✓ Label - Display text on the screen

- ✓ Button - Contain text and can perform an action when clicked
- ✓ Entry -Allows only a single line of text
- ✓ Text -Allows multiline text entry
- ✓ Frame -rectangular region used to group related widgets or provide padding between widgets

5.1.Implementation of MYSQL

MySQL is an open source relational database management system (RDBMS) that can be easily implemented and managed either on-premise or via the cloud through a hosting provider. It supports lots of simultaneous writes and scales via replication.

5.2.INSERTING AN ITEM :

In this code, we have to insert the items into the stock or store system. When we insert the items only the persons can buy the product or can sale the product. After inserting the we have to the item id or item number, item name, item details and the price of the product. Then only we can display the product details. After inserting the product, we have to give clear option to store the data what we insert in the stock trade or eco live cart. In the stock trade we can insert any products like pesticides,insceticides,fertilizersetc.,

def _add_btn_clickked(self):

#print("Clicked")

itno = int(self.entry_1.get()) amount = int(self.entry_11.get()) if itno==0 and amount==0:

tm.showerror("Entry Error", "Fill Your all fields") else:

cur.execute ("""UPDATE shopping1 SET amt='PAID' WHERE itno='%d' ""% (itno)) conn.commit() tm.showinfo("Confirmation","PaymentSuccesful")

5.2.DELETING AN ITEM :

For deleting the items, wecan use delete function or removing the items from thestore.Whenwedeletetheitemoritsalefromthestoreitwilldeleteoritdisplays aspaid.Whenwedeletetheitemfromthestore,thedetailsoftheitemwillbeerased or deleted from the database. Once we delete the products from the cart, it displays a message box like the was deletedsuccessfully.

```
def delete(self):

chil=tree.get_children()

for i in chil:

tree.delete(i)

tm.showinfo("Data Cleared","Data has been Cleared")

def delete1(self):

chil=tree.get_children()

for i in chil:

tree.delete(i)
```

5.3.PAYMENT METHOD :

After adding the items into the cart, we have to buy the product from the storebyusingonlinemethodlikedebitcardorcreditcard.Forpaying,wehavetoinsertthe carddetailslikecardnumber,cardname,cvv,carddate.Afteraddingthecarddetails, we have clicked the payment button then only the item will sale or not. Once done with the payment we cannot change any item or product details. After that we have use clear button, then only the item will be displayed as the product is saled. Once we done the payment process, it displays a message box like payment successful. After displaying the message box only we can get the paymentmessage.

```
def _add_btn_clickked(self):

#print("Clicked")
```

itno = int(self.entry_1.get()) amount = int(self.entry_11.get())

if itno==0 and amount==0:

tm.showerror("Entry Error", "Fill Your all fields")

else:

cur.execute ("""UPDATE shop4 SET amt='PAID' WHERE itno='%d' """ % (itno)) conn.commit()

tm.showinfo("Confirmation","PaymentSuccesful")

5.4.DISPLAYING ITEMS :

For displaying the items, we have to use the display function .For displaying all the details of the product what we insert into the cart. It displays only the cart details what items will be stored in the cart or any stock. It displays all the items details like item number, itemdetails, itemname and itemamount etc., once we display the item from the data. It displays a message like the item was displays successfully from thecart.

def _view_btn_clickked(self):

tree["columns"] = ("one", "two", "three","four","five","six") tree.column("one", width=100)

tree.column("two", width=100) tree.column("three", width=100)

tree.column("four", width=100)

tree.column("five", width=100)

tree.column("six", width=100)

tree.heading("#0", text='S.NO')

tree.heading("one", text="ITEM CODE") tree.heading("two", text="ITEM NAME")

tree.heading("three", text="QUANTITY")

tree.heading("four", text="RATE")

tree.heading("five", text="TOTAL AMOUNT") tree.heading("six",text="PAID STATUS")

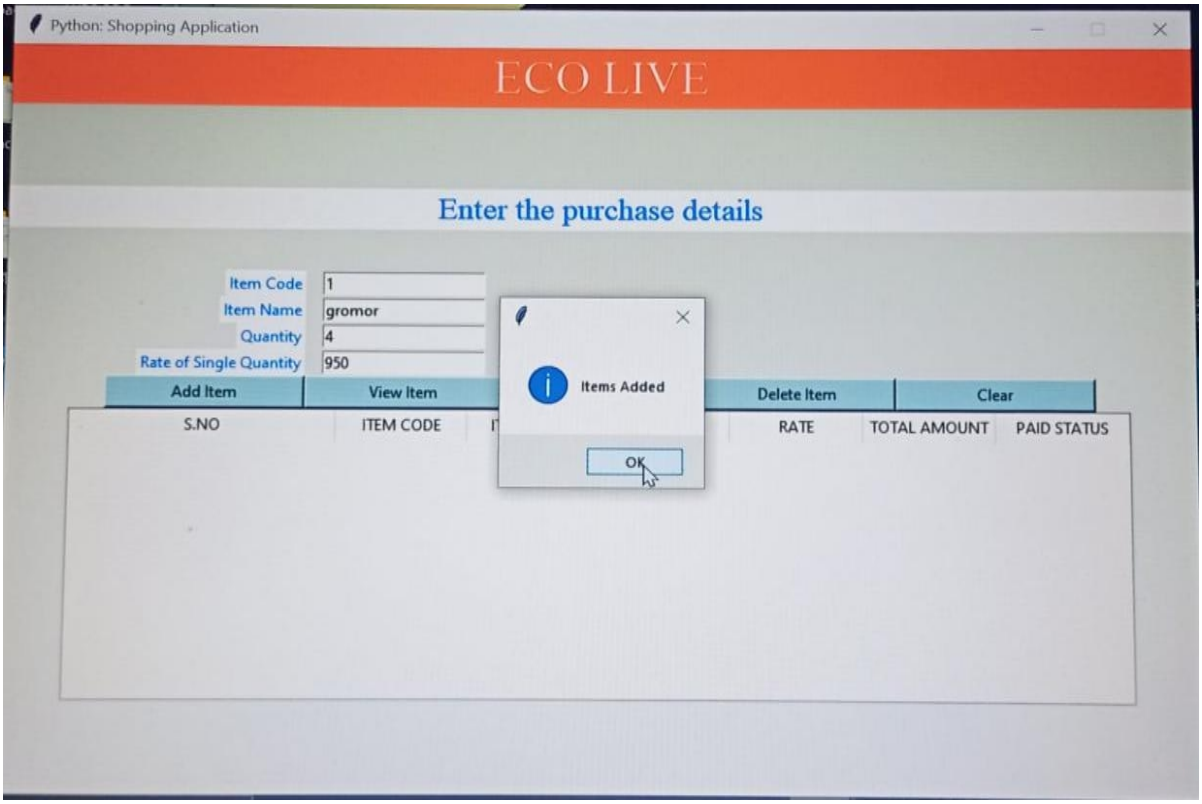
```
cur.execute("select * from shop4;")

cpt = 0 # Counter representing the ID of your code.

for row in cur:

tree.insert('','end',text=str(cpt),values=(row[0],row[1],row[2],row[3],row[4],row[5]), tags

=('oddrow','evenrow'))

tree.tag_configure('oddrow',background='orange')

tree.tag_configure('evenrow', background='white')

cpt += 1 # i
```

CHAPTER 6

RESULTS

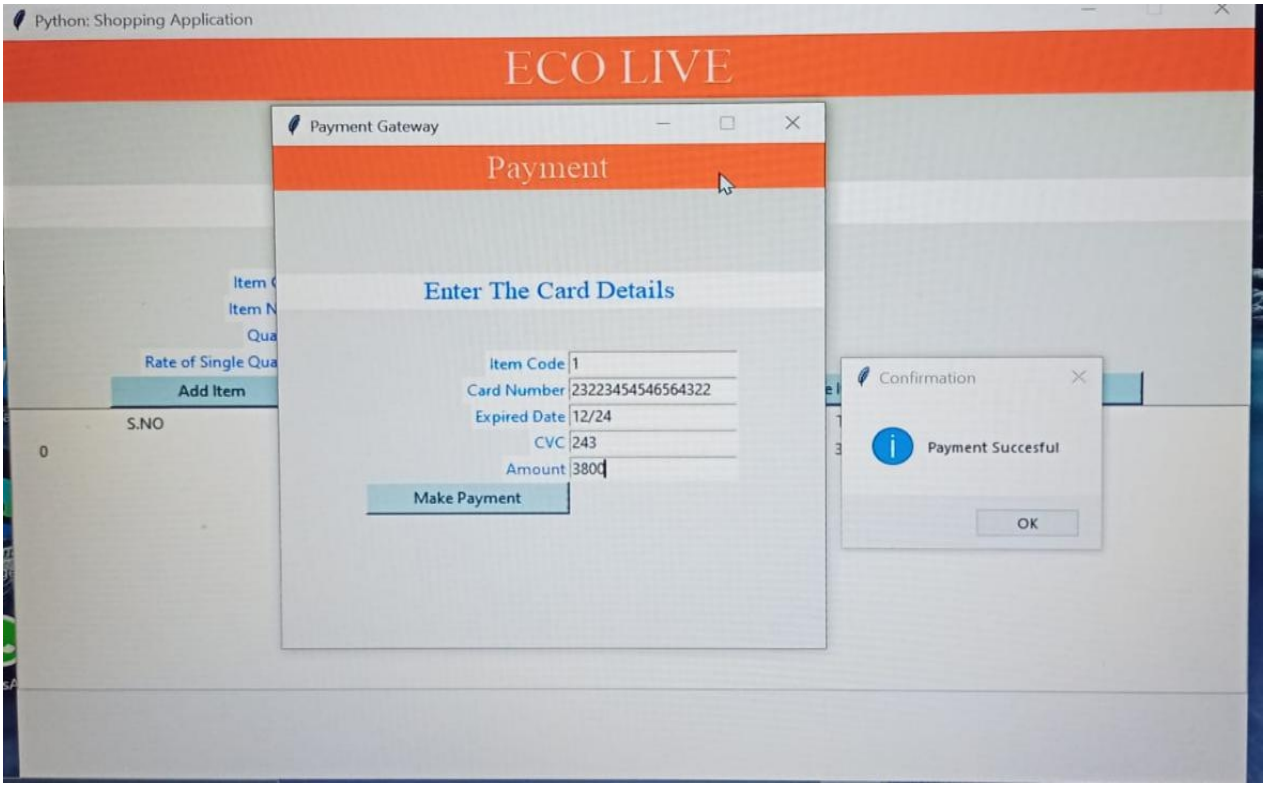
6.1 Entering details

This is the output screen displayed after running the code. Here we can add item by entering item number, item name, quantity, rate of single quantity and after entering the details click on add item.



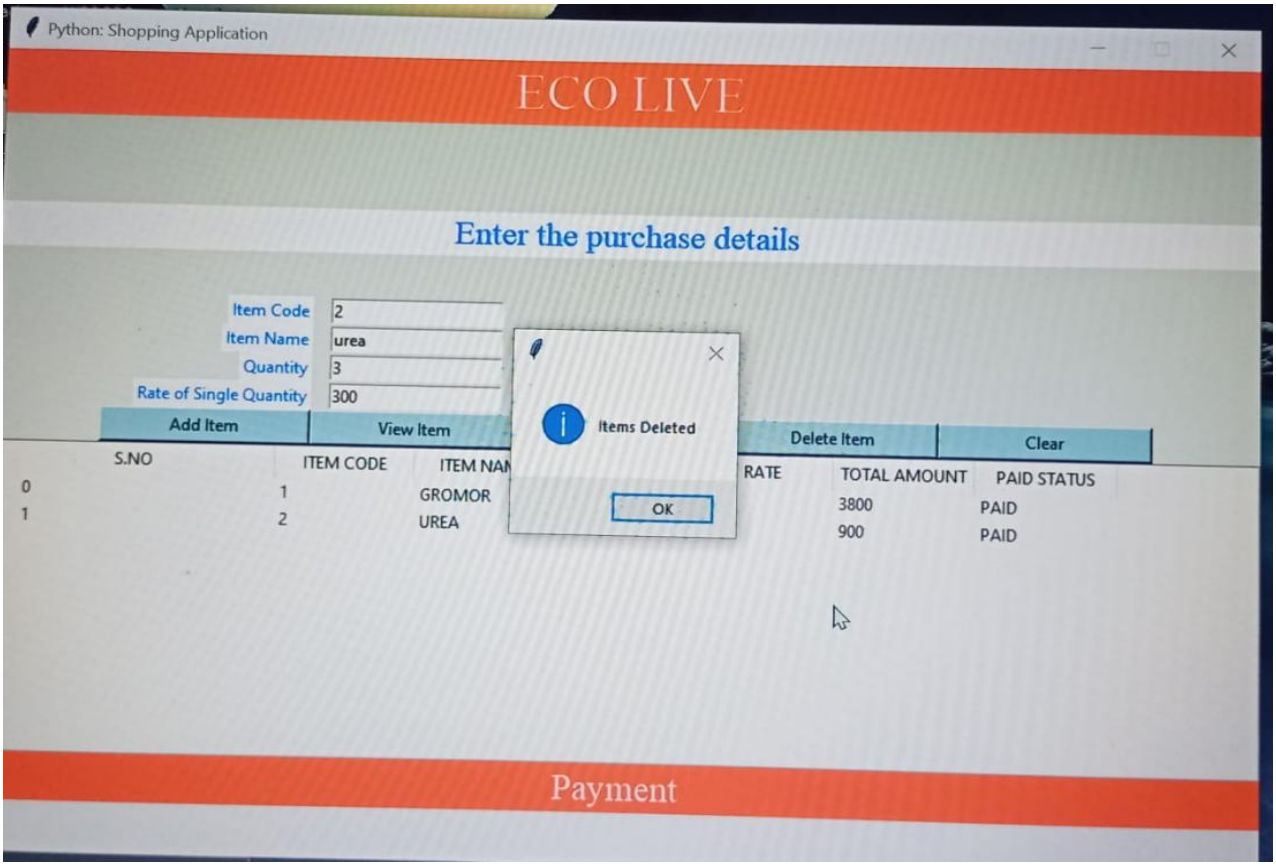
6.2 After adding details

After entering the details we have to click on add button and after selecting the add button the dialogue box item added is displayed.



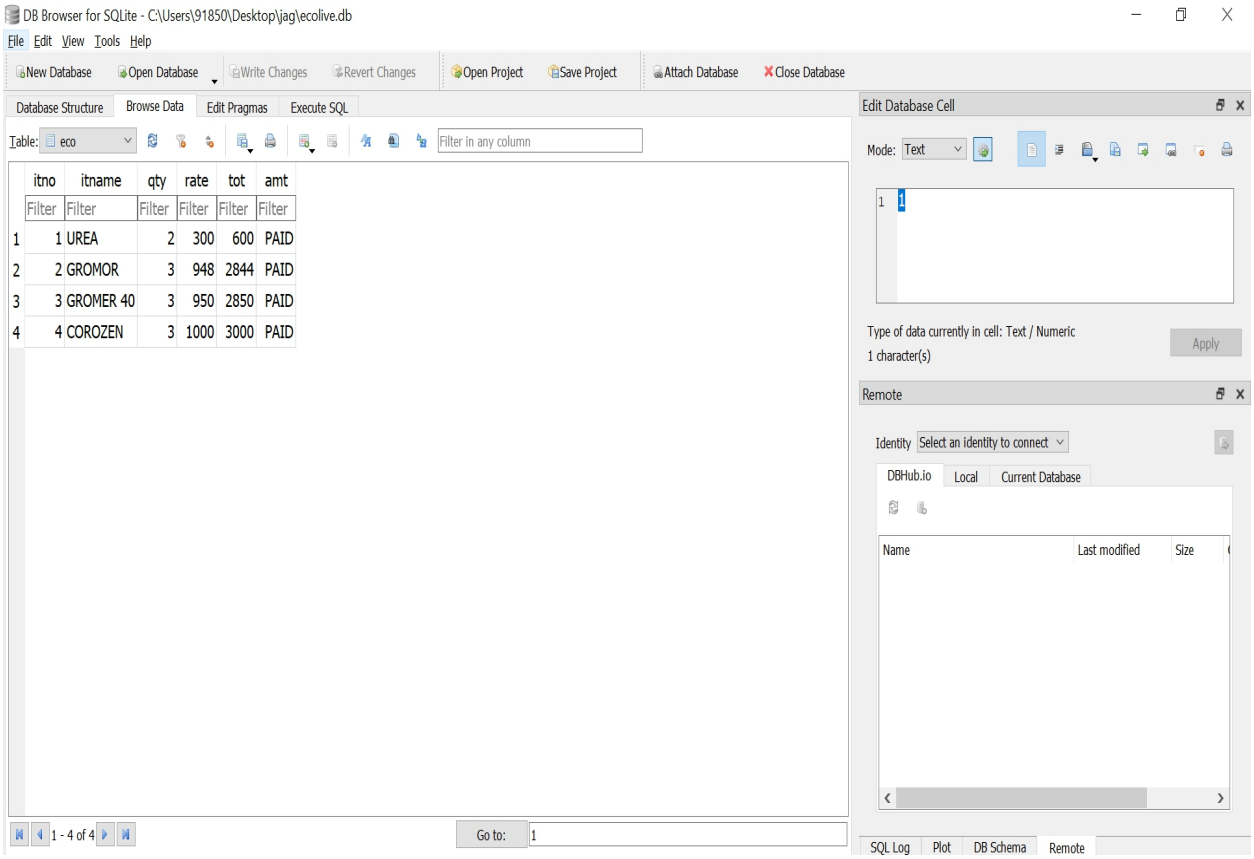
6.3 Payment details

After adding we can make payment by selecting the payment button, and after giving the required card details for payment the payment is done.



6.4 Deleting the items

We can delete the item by selecting the delete option and we have to give the item number for deleting the item.



6.5 Data stored in MySQL

Here the data is stored in database, when we delete the item or add or make the payment the data is automatically updated in the database.

CONCLUSION

Python is a high-level, interpreted scripting language developed in the late 1980s by Gui do vanRossum at the National Research Institute for Mathematics and Computer Science in the Netherlands. The initial version was published at the alt. sources newsgroup in 1991, and version 1.0 was released in1994.

Python is still maintained by a core development team at the Institute, and Guido is still in charge, having been given the title of BDFL (Benevolent Dictator for Life) by the Python community. The name Python, by the way, derives not from the snake, but from the British comedy troupe Monty Python's Flying Circus, of which Guidowas, and presumably still is, a fan. It is common to find references to Monty Python sketches and movies scattered throughout the Python documentation.

We can able to appreciate the significant change in people's behavior in many ways like they buying pattern, attitude after going through a vast study on shopping. Benefits of the ECO LIVE is it can be done by the low start-up co stand quick to setup. Developing ECO LIVE is inexpensive. It helps in effective solution for a limited amount of item. It requires less custom development and easy handling the data of the cart in detail at any time. This cart can run on any server platform. This project relates the python concept in the stock trading items. In real life, stock trade is very important. Buying products from online or offline clothes, electronic devices or something else we bought from the shops only. If we interest to the product we will buy the product in the manner of payment method.

REFERENCES

Database systems Models, Languages, Design and Application Programming, RamezElmasri and Shamkant B. Navathe, 7th Edition, 2017

Database management systems, Ramakrishnan, and Gehrke, 3rd Edition, 2014, McGraw Hill

StackOverflow:www.stackoverflow.com

www.tutorailspoint.com

https://www.geeksforgeeks.org/python-gui-tkinter/

https://docs.python.org/2/library/tkinter.html

www.youtube.com

Python GUI Programming Cookbook

www.wikipedia.org

www.dev.mysql.com/doc/