**NAME: P LAKSHMI SUMANA**

**USN: 1NH18CS13**

**SEM: 5**

**SEC: B**

**TITLE: PARKING SLOTS**

# CHAPTER 1

## INTRODUCTION

### 1.1 PROBLEM DEFINATION

The main objective of this mini project is to develop an application that will have the following functions: -

- Basically, mini project helps us to explore and strengthen the understanding of fundamentals through practical application of theoretical concepts.
- It also helps us to boost your skills and widen your horizon of thinking.
- It helps the beginners to do larger projects in their career.
- It is helpful to design our algorithm.
- Better learning of the coding language.
- To implement the concepts and learn to implement them properly.

### 1.2 OBJECTIVES

The main aim of the project is to book the slots of parking a vehicle for required time where fair charges are pressed for allotment as to avoid the mess in parking lot.  It manages the details of duration, vehicles, parking slots and parking fares. It manages all the information about parking duration, slots and parking fare. The project is totally built at administrative end and thus only the administrator is guaranteed the access. The purpose of the project is to build an application program to reduce the manual work for managing the things. There will be a separate category for faculty and students to park the vehicle.

## 1.3 METHODOLOGY TO BE FOLLOWED

The main objectives of this project are:
1. It helps us to book a slot for parking vehicles.
2. It makes easier so we can get quick services as soon as possible
3. To keep the information of the customer
4. It is used to record the details of various activities of the user
5. The main objective is to achieve high level of customer satisfaction by providing efficient service.

## 1.4 EXPECTED OUTCOMES

The output of the system is displayed on the console during a formatted manner. The output may attend a file (file style already explained).

➢ Main page contains login our sign
➢ First sign up by filling in required details
➢ Logging in with registration id and password
➢ Display the details of the parking slots
➢ Fill in few details to book a slot
➢ Display the amount to be paid

## CHAPTER 2:

## REQUIREMENT SPECIFICATIONS:

### 2.1 HARDWARE REQUIREMENTS

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

## 3.1. PYTHON FUNDAMENTALS:

**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.

**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 fundamentals consist of basic building blocks of Python programming language. And it is basically divided into the following categories.

1.    Statements

2.    Indentations

3.    Comments

4.    Variables

5.    Constants

6.    Tokens

**1. Statements:** They are logical instructions that interpreter can execute and read; it can also be both single and multiline.

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

**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. Comments:**

Comments are basically nothing but tagged lines of in codes which inceases 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.

**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.

**5. Constants:**

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

**6. Token:**

In Python program the 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 programming identifiers are nothing but user-defined names to the represent all programmable entity like variables, functions, modules, classes. There are few rules that we need to following 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

**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 available 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:**

List box 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>>.

## 3.2. 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 removes 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), and 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

### ER DIAGRAM:
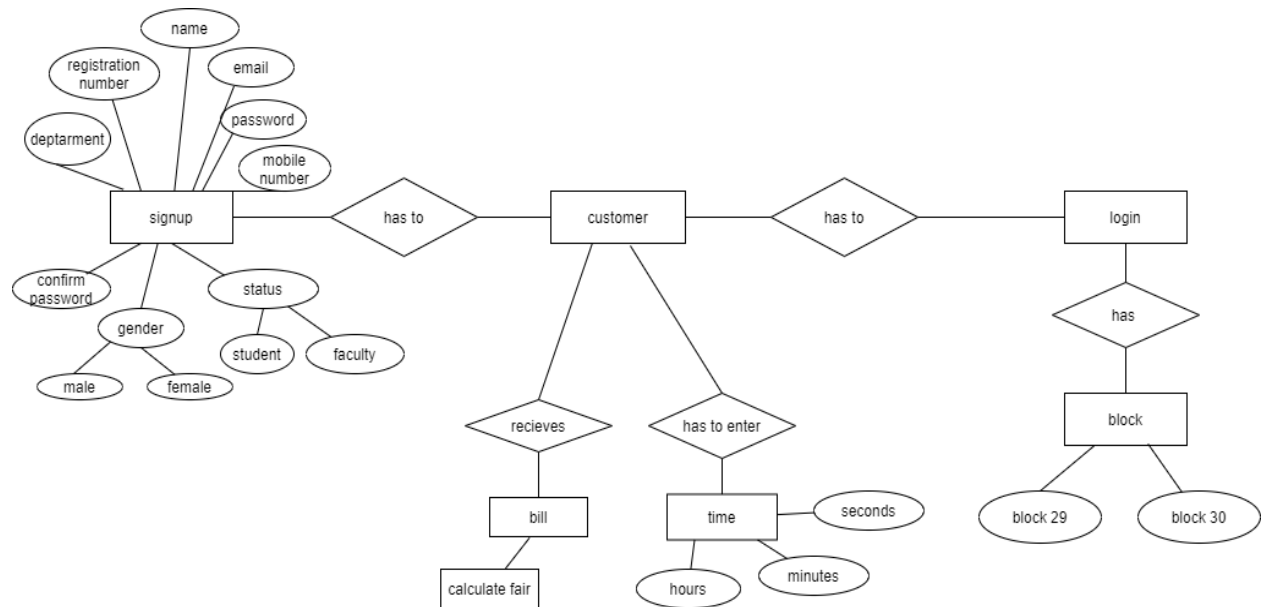
# CHAPTER 5
## IMPLEMENTATION

Implementation of Function

- A function is a block of code which only runs when it is called.

- We can pass data, known as parameters, into a function.

- "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

- o Tkinter is the standard GUI library for Python.
- o fast and easy way to create GUI applications.
- o 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

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.

## 5.1.   INSERTING DETAILS:

In this code, we have to insert the details of customer and product into the stock or add. When we insert the details only. After inserting the we have to the customer name, customer gender, phone number, date, location, product name and cost. Then only we can display the details. After inserting the details, we have to give clear option to store the data what we insert in the stock or interior product cart. In the stock we can insert any products like marbles, tiles, sofa etc.,

## 5.2.   DISPLAYING DETAILS:

For displaying the details, we have to use the display function. For displaying all the details of the product details with customer details 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 customer name, customer gender, phone number, date, location, product name and cost., once we display the item from the data. It displays a message like the item was displays successfully from the cart.

## 5.3.   DELETE DETAILS:

For deleting the details, we can use delete function or removing the items from the store with the help of customer name. When we delete the item from the store it will delete When we delete the item from the store, the details of the item will be erased or deleted from the database. Once we delete the products from the cart, it displays a message box like the was deleted successfully.

## 5.4.   ADMIN LOGIN:

For displaying the details, we have to use the display function. For displaying all the details of the product details with customer details 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 customer name, customer gender, phone number, date, location, product name and cost.

## 5.5. UPDATE DETAILS:

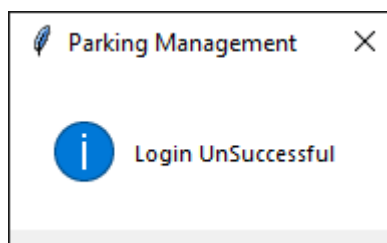Updating the details with the help of customer name, then displaying all the details.

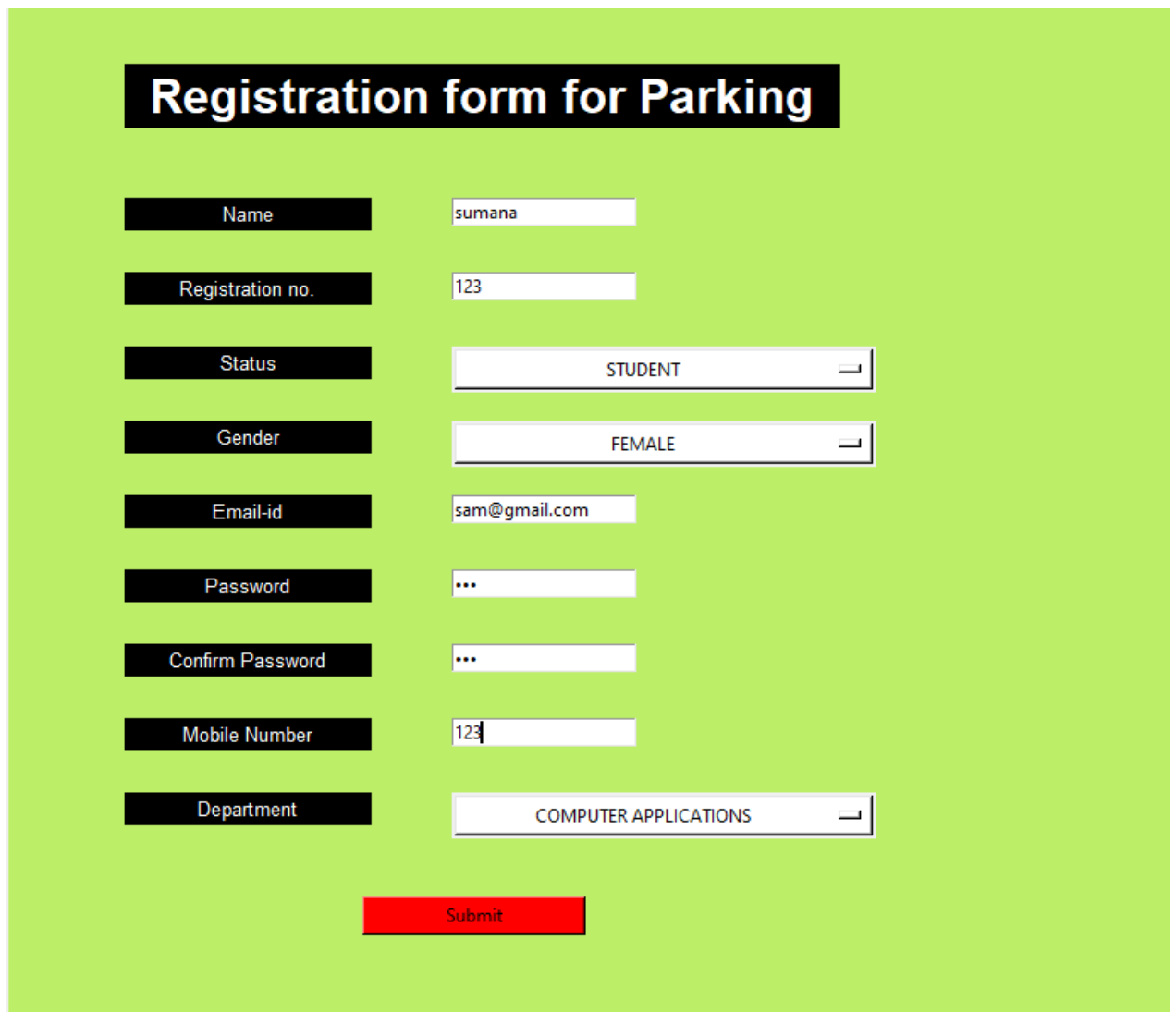# CHAPTER 6

## RESULTS

### Interpretation of Results

Result 5.1



This is the login-page. Once the user registers his details in the sign-up page, the login-page pops up and he has to enter the same details in the login-page or the error page occurs.

Result 5.2



This is the sign-up page for the user on our GUI.once the user enters the username and password, the details will be stored in the database.

Result 5.3



Once the users log-in the application this window pops-up. It contains all the database operations that are needed to be done.

Result 5.4

Once we give all the given data which must be filled by the user. And press the key of calculate fare it shows the total fare.

Result 5.5



Once we are done by given all the data and see the total fare.

# CHAPTER 7

## CONCLUSION

This project mainly deals with the catchupcab which is designed using python gui application. It has its own merits and demerits. The main advantage of this project is that it is easy to book a cab through our mobile applications. Therefore the project has been designed to provide the customer with proper convenient whether the cabs are in proper convenient or not.

# 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/