KENDRIYA VIDYALAYA

OCF AVADI

INVENTORY PROJECT ON

Parking management system



SUBMITTED BY: SUBMITTED TO:

J.GOPIKA RANI(XII-B) MS.ANJU RANI

D.PRIYANGA(XII-B) PGT

(COMP.SCI)

**INDEX**

|  |  |  |
| --- | --- | --- |
| S.NO | TITLE | PAGES |
| 1. | Bonafide certificate | 3 |
| 2. | Acknowledgement | 4 |
| 3. | Main characteristics of this project | 5 |
| 4. | PYTHON | 6-7 |
| 5. | MYSQL | 8-10 |
| 6. | Source code | 11-15 |
| 7. | Output | 16 |

**Bonafide Certificate**

This is a bonafide record of the project titled **Parking**

**Management System** done by **J.Gopika Rani , D.Priyanga**

**during** the year of **2019-2020**.

Submitted in partial fulfillment of the requirement for the practical Examination for the XII in the academic year

of 2019-2020

**ROLL NO:**

**DATE:**

**INTERNAL EXAMINER EXTERNAL EXAMINER**

**PRINCIPAL**

**SCHOOL SEAL:**

**Acknowledgement**

we would like to express our special thanks of gratitude to our teacher **Ms. ANJU RANI PGT (COMP.SCI)** as well as our principal **Mr.P.SAKTHIVEL** who gave us the golden opportunity to do this wonderful project on the topic **Parking Management System** which alsohelp us in doing a lot of research and came to know about so many new things we are really thankful to them.

Secondly we would also like to thank our gods, parents and friends who helped us a lot in finalizing this project within the limited time framed.

**MAIN CHARACTERSTICS OF THIS PROJECT**

* Parking management system for managing the records of the incoming and out going vehicles in an parking house
* It’s an easy for Admin to retrieve the data if the vehicle has been visited through number

he can get that data.

* Now days in many public places such as malls, multiplex system, hospitals, offices ,market areas there is a crucial problem of vehicle parking.
* Parking control system has been generated in such a way that it is filled with many secure devices such as, parking control gates, toll gates, time and attendance machine ,car counting system etc
* The objective of this project is to build a Vehicle Parking management system that enables the time management and control of vehicles using number plate recognition.
* The system that will track the entry and exit of cars, maintain a listing of cars within the parking lot, and determine if the parking lot is full or not.
* It will determine the cost of per vehicle according to their time consumption.

**PYTHON**

* Python is an interpreted, object-oriented, high-level programming language with dynamic semantics.
* Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together.

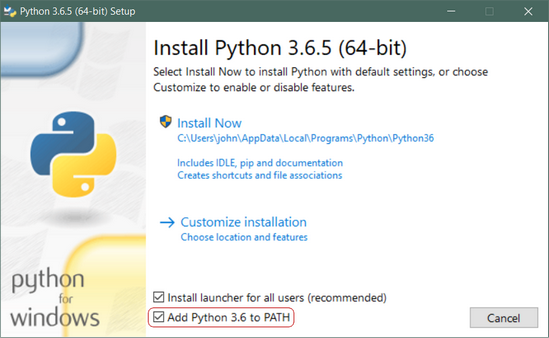
PYTHON INSTALLATION PROCESS:

Step 1: Download the Python 3 Installer

1. Open a browser window and navigate to the [Download page for Windows](https://www.python.org/downloads/windows/) at [python.org](https://www.python.org/).
2. Underneath the heading at the top that says **Python Releases for Windows**, click on the link for the **Latest Python 3 Release - Python 3.x.x**. (As of this writing, the latest is Python 3.6.5.)
3. Scroll to the bottom and select either **Windows x86-64 executable installer** for 64-bit or **Windows x86 executable installer** for 32-bit.

### Step 2: Run the Installer

Once you have chosen and downloaded an installer, simply run it by double-clicking on the downloaded file. A dialog should appear that looks something like this:

**[](https://files.realpython.com/media/win-install-dialog.40e3ded144b0.png)**

**Mysql**

* MySQL is an Oracle-backed open source relational database management system ([RDBMS](https://searchdatamanagement.techtarget.com/definition/RDBMS-relational-database-management-system)) based on Structured Query Language ([SQL](https://searchsqlserver.techtarget.com/definition/SQL)).
* MySQL runs on virtually all platforms, including [Linux](https://searchdatacenter.techtarget.com/definition/Linux-operating-system), [UNIX](https://searchdatacenter.techtarget.com/definition/Unix) and

[Windows](https://searchwindowsserver.techtarget.com/definition/Windows).

**MySQL installation process:**

**Step 1: download MySQL**  
Download MySQL from [dev.mysql.com/downloads/](http://dev.mysql.com/downloads/). Follow MySQL Community Server, Windows and download the “Without installer” version.

**Step 2: extract the files**  
We will install MySQL to C:mysql, so extract the ZIP to your C: drive and rename the folder from “mysql-x.x.xx-win32” to “mysql”.MySQL can be installed anywhere on your system. If you want a lightweight installation, you can remove every sub-folder except for bin, data, scripts and share.

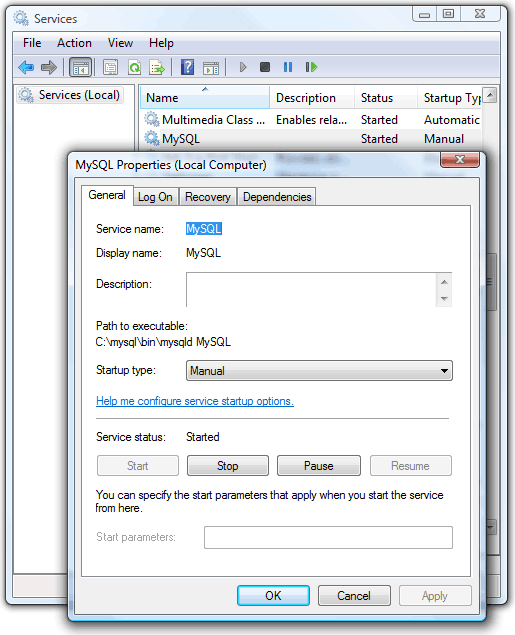
**Step 3: create a configuration file**  
MySQL provides several configuration methods but, in general, it is easiest to to create a my.ini file in the mysql folder.

**Step 4: test your installation**  
The MySQL server is started by running C:mysqlbinmysqld.exe. Open a command box (Start > Run > cmd).This will start the MySQL server which listens for requests on localhost port 3306. You can now start the MySQL command line tool and connect to the database.

**Step 5: change the root password**  
The MySQL root user is an all-powerful account that can create and destroy databases. If you are on a shared network, it is advisable to change the default (blank) password.

**Step 7: Install MySQL as a Windows service**  
The easiest way to start MySQL is to add it as a Windows service. Open the Control Panel, Administrative Tools,

then Services and double-click MySQL. Set the Startup type to “Automatic” to ensure MySQL starts every time you boot your PC.



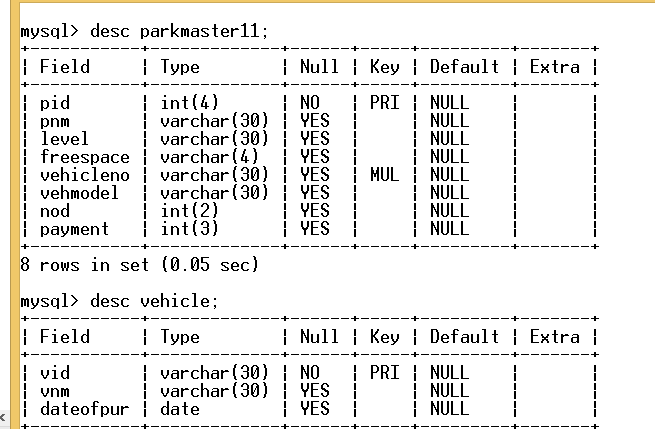
# PARKING MANAGEMENT SYSTEM

DBMS : MYSQL

HOST NAME : LOCAL HOST USER : ROOT

PASSWORD: “a” DATABASE: pms

# IMAGES OF TABLES:



Python Code:

'''

import os import platform

import mysql.connector mydb=mysql.connector.connect(host="localhost",user="root",password="a",database="pms") mycursor=mydb.cursor()

def Add\_Record(): L=[]

id1=int(input("Enter the parking number : ")) L.append(pid1)

pname1=input("Enter the Parking Name: ") L.append(pname1)

level1=input("Enter level of parking : ") L.append(level1)

freespace1=input("Is there any freespace or not :YES/NO ") L.append(freespace1)

vehicleno1=input("Enter the Vehicle Number : ") L.append(vehicleno1)

nod1=int(input("Enter total number of days for parking: ")) L.append(nod1)

Payment1=int(input("Enter total payment : ")) L.append(payment1)

stud=(L)

sql="insert into parkmaster11 (pid1,pnm1,level1,freespace1,vehicleno1,nod1,payment1) values (%s,%s,%s,%s,%s,%s,%s)"

mycursor.execute(sql,stud) mydb.commit()

def RecView():

print("Select the search criteria : ") print("1. Parking Number") print("2. Parking Name")

print("3. Level No") print("4. All")

ch=int(input("Enter the choice : ")) if ch==1:

s=int(input("Enter Parking no : ")) rl=(s,)

sql="select \* from parkmaster11 where pid1=%s" mycursor.execute(sql,rl)

elif ch==2:

s=input("Enter Parking Name : ") rl=(s,)

sql="select \* from parkmaster11 where pnm1=%s" mycursor.execute(sql,rl)

elif ch==3:

s=int(input("Enter Level of Parking : ")) rl=(s,)

sql="select \* from parkmaster11 where level1=%s" mycursor.execute(sql,rl)

elif ch==5:

sql="select \* from parkmaster11" mycursor.execute(sql) res=mycursor.fetchall()

print("Details about Parking are as follows : ")

print("(Parking Id,Parking Name,Level,FreeSpace(Y/N),Vehicle No,No of days for parking,Payment)") for x in res:

print(x)

def Vehicle\_Detail():

L=[]

vid1=int(input("Enter Vehicle No : ")) L.append(vid1)

vnm1=input("Enter Vehicle Name/Model Name : ") L.append(vnm1)

dateofpur1=input("Enter Date of purchase : ") L.append(dateofpur1)

vdt=(L)

sql="insert into vehicle (vid1,vnm1,dateofpur1) values (%s,%s,%s)" mycursor.execute(sql,vdt)

mydb.commit() def Vehicle\_View():

vid=int(input("Enter the vehicle number of the vehicle whose details is to be viewed : "))

sql="Select parkmaster11.pid, parkmaster11.pnm, parkmaster.level, vehicle.vid,vehicle.vnm,vehicle.dateofpur from parkmaster11,vehicle where parkmaster11.vid=vehicle.vid "

rl=(vid,) mycursor.execute(sql,rl) res=mycursor.fetchall() for x in res:

print(x) def remove():

vid1=int(input("Enter the vehicle number of the vehicle to be deleted : ")) rl=(roll,)

sql="Delete from vehicle where vid1=%s" mycursor.execute(sql,rl)

mydb.commit() def Menu():

print("Enter 1 : To Add Parking Detail") print("Enter 2 : To View Parking Detail ") print("Enter 3 : To Add Vehicle Detail ") print("Enter 4 : To Remove Vehicle Record") print("Enter 5 : To see the details of Vehicle")

input\_dt = int(input("Please Select An Above Option: ")) if(input\_dt== 1):

Add\_Record() elif (input\_dt==2):

RecView()

elif (input\_dt==3): Vehicle\_Detail() elif (input\_dt==4):

remove()

elif (input\_dt==5): Vehicle\_View()

else:

print("Enter correct choice. . . ") Menu()

def runAgain():

unAgn = input("\nwant To Run Again Y/n: ") while(runAgn.lower() == 'y'):

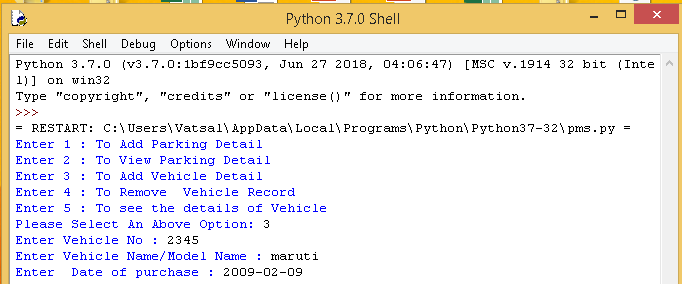
if(platform.system() == "Windows"): print(os.system('cls'))

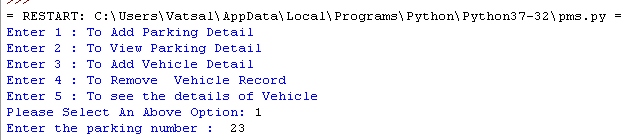
else:

print(os.system('clear')) MenuSet()

runAgn = input("\nwant To Run Again Y/n: ") runAgain()

**OUTPUT**





**THANK YOU**

**J.GOPIKA RANI (XII-B)**

**D.PRIYANGA (XII-B)**