INDIAN SCHOOL AL GHUBRA

CLASS XII COMPUTER SCIENCE 2024-2025

LAB ASSIGNMENT 1

Write a user defined function that takes two integer numbers and an arithmetic operator as arguments/parameters and returns the result of the arithmetic operation. The valid arithmetic operators are +, -, \*, / and %. The function s should return “Invalid operator” if the operator is other than +, -, \*, / and %. Also write a main program to invoke this function.

LAB ASSIGNMENT 2

Write a user defined function that takes an integer number as argument and returns the factorial of that number. Also write a main program to invoke this function.

LAB ASSIGNMENT 3

Write a user defined function that takes a string and an empty dictionary as arguments and the function stores the characters of the string as keys and frequency of its occurrence as value to the dictionary. Also write a main program to invoke this function.

LAB ASSIGNMENT 4

Gianluca has started to create a program to analyze pupil attendance. He wants to use a menu system with subprograms.

The program must include subprograms to:

• Display the names of students whose percentage attendance was less than 75%

• Count and display the number of students whose percentage attendance was 90% or higher.

Complete the program code to implement these requirements using the structure given below.

# write subprogram for menu option 1 here

# write subprogram for menu option 2 here

# menu program

pupilAttendance = [["Faroukh" , "Salah" , 70],

["Kelvin", "Glintode" , 85],

["Lara" , "Godfrey" , 90],

["Amara" , "Grzinski" , 70],

["Aaron" , "Grimshaw" , 90],

["Farihaa" , "Mohan" , 95],

["Taz" , "Grimstow" , 60],

["Ali" , "Aisha" , 95],

["Charlene" , "Hall" ,85],

["Asra" , "Ashiq" , 90],

["Sadia" , "Bhatti" , 65],

["Ria" , "Hall" , 90],

["Fernado" , "Askabat" , 60],

["Richard" , "Hawkins" , 80],

["Siyao" , "Wang" , 60],

["Marketta" , "Hosier" , 100]]

option = 0

print ("Attendance Menu Options")

print ("1 - Display names of low attendance")

print ("2 - Display number of high attendance")

option=int(input("Choose option: "))

print ('\n')

if option == 1:

# complete the if statement

elif option == 2:

# complete the else if statement

else:

print("Program complete")

LAB ASSIGNMENT 5

Robert runs an online order delivery system.

He wants a program to check product codes stored in a file. The file details.txt contains the list of product and product codes.

The program must:

i. product and product codes are separated from @ sign

ii. check each product code to ensure that:

1. the first three letters of the product code are first three letters of the product.

2. if product code is also including at least one digit (0–9)

3. if a product code does not meet these requirements:

display the product code and write those product codes in a new file called, “Errors.txt” file.

4. If a product code does meet the requirements: display the product code and write those product codes in a new file called, “Finalcodes.txt” file.

details.txt

PENCIL@PENCIL001

BALLPEN@BALL

MICROTIPPEN@MICRO001

HIGHLIGHTER@PEN003

PLASTICSCALE@SCALE002

STEELSCALE@STEELSCALE123

DRAWINGPINS@DRAWINGPINS23

PENCIL@PENCIL004

BALLPEN@BALL002

MICROTIPPEN@MICR##

HIGHLIGHTER@PEN004

PLASTICSCALE@SCALE003

STEELSCALE@STEELSCALE124

DRAWINGPINS@DRAWINGPINS24

LAB ASSIGNMENT 6

**Email.txt** contains a list of email addresses.

Write an UDF EMAILCHECK() to implement these requirements.

The code must:

• check each email address to ensure it contains the ‘@’ symbol.

• write email addresses that do not contain the ‘@’ symbol to an **Error.txt** file.

Write an UDF COUNTUK(filename) to implement these requirements.

The code must:

* Ensure that the function take filename as the argument
* Count and display the emails ending with uk

Email.txt

Ursa.Collins.@anuverplaice.co.uk

George.Green.+anuverplaice.com

Leanne.Young.@newmail1919.co.uk

Mary.Ulrich.%sumwere19.com

Zane.Quaid.&myschul2018.co.uk

Nancy.Oakley.@anuverplaice.com

Aaron.Zachary.@somewer2000.co.uk

Francis.Ulrich.%newmail1919.co.uk

Thomas.Adams.@newmail1919.co.uk

Zane.Adams.\*myschul2018.co.uk

Zane.Pearson.@anuverplaice.co.uk

Wilma.Lawrence.%anuverplaice.com

Zane.Montgomerie.@anuverplaice.co.uk

Victoria.Drew.+newmail1919.co.uk

Mary.Rawson.@newmail1919.co.uk

Mary.Winston.@newhere2019.com

Wilma.Rawson.@myschul2018.co.uk

Thomas.Evans.@mycoleg2010.co.uk

Yvette.Turner.@sumwere19.com

Aaron.Valdez.%anuverplaice.co.uk

Jason.Pearson.@sumwere19.com

Yvette.King.@mycoleg2010.co.uk

Thomas.Green.@newmail1919.co.uk

Nancy.Turner.&sumwere19.com

Mary.Green.%sumwere19.com

Karla.Turner.@anuverplaice.com

Colin.Oakley.@newhere2019.com

Thomas.Xiong.@newmail1919.co.uk

Ursa.King.@mycoleg2010.co.uk

Leanne.Zachary.@newhere2019.com

Zane.Lawrence.@sumwere19.com

Zane.Oakley.@myschul2018.com

Xavier.Ulrich.@myschul2018.co.uk

Steven.Zachary.@mycoleg2010.co.uk

Ursa.Collins.@newmail1919.co.uk

Colin.Nelson.@somewer2000.co.uk

Colin.Turner.@newmail1919.co.uk

Olive.Rawson.@newhere2019.com

Yvette.Hiatt.+somewer2000.co.uk

Jason.Forrest.\*mycoleg2010.co.uk

Peter.Winston.@mycoleg2010.co.uk

Ursa.Montgomerie.%mycoleg2010.com

Karla.Winston.&myschul2018.com

Everett.Forrest.+myschul2018.co.uk

Bonnie.Young.%somewer2000.co.uk

Yvette.Nelson.@newmail1919.co.uk

Steven.Nelson.\*myschul2018.com

Diana.Rawson.@sumwere19.com

Jason.Valdez.@anuverplaice.com

George.Nelson.@myschul2018.com

Wilma.Rawson.@myschul2018.co.uk

Diana.Pearson.@anuverplaice.co.uk

Olive.Green.&myschul2018.com

Bonnie.Oakley.@mycoleg2010.com

Ursa.Jones.+newhere2019.com

Peter.Ulrich.\*mycoleg2010.co.uk

Victoria.Hiatt.@myschul2018.co.uk

Everett.Green.@myschul2018.co.uk

George.Jones.@newmail1919.co.uk

Olive.Quaid.@mycoleg2010.com

Wilma.Pearson.@sumwere19.com

Leanne.Hiatt.\*anuverplaice.co.uk

Wilma.Montgomerie.%anuverplaice.co.uk

Karla.Quaid.@mycoleg2010.co.uk

Diana.Lawrence.@mycoleg2010.co.uk

Zane.Hiatt.@myschul2018.com

Zane.Zachary.&myschul2018.com

Steven.Forrest.&anuverplaice.com

Peter.Smith.@mycoleg2010.com

Xavier.Ulrich.&mycoleg2010.com

Zane.Xiong.@newhere2019.com

Aaron.Forrest.@newhere2019.com

Aaron.Quaid.&newhere2019.com

Olive.Adams.@myschul2018.com

Leanne.Turner.+newhere2019.com

Colin.Smith.@somewer2000.co.uk

Yvette.Smith.@mycoleg2010.co.uk

Randall.Young.&newhere2019.com

Francis.Forrest.+anuverplaice.com

Steven.King.@mycoleg2010.co.uk

Leanne.Forrest.@myschul2018.com

Leanne.Young.@mycoleg2010.co.uk

Randall.Young.@somewer2000.co.uk

Ursa.Quaid.@myschul2018.com

Peter.Drew.@anuverplaice.com

Leanne.Rawson.@newmail1919.co.uk

Randall.King.@mycoleg2010.co.uk

Peter.Xiong.&mycoleg2010.com

Bonnie.Oakley.@myschul2018.com

Harriett.Turner.@mycoleg2010.co.uk

Everett.King.@myschul2018.com

Yvette.Zachary.@mycoleg2010.co.uk

Zane.Brown.@somewer2000.co.uk

Xavier.Winston.@anuverplaice.com

Zane.Zachary.@sumwere19.com

Karla.Pearson.@myschul2018.co.uk

Thomas.Jones.\*sumwere19.com

Steven.Green.@myschul2018.com

Leanne.Zachary.@anuverplaice.com

Nancy.Drew.@myschul2018.com

LAB ASSIGNMENT 7

Farshia is the regional manager for an insurance company.

She manages a team of sales staff.

She wants a program to analyse the performance of her team over a number of

months.Sales details are stored in a text file Sales.txt

Write a program to:

• calculate and display the total sales made by each member of the team

• calculate and display the total sales made by the whole team.

# Structure of sales record is

# StaffID, First name, Last name, January sales, February sales,

# March sales, April sales, May sales, June sales

Sales.txt

101TGY,George,Taylor,6009,5262,3745,7075,1943,4432

103FCY,Fehlix,Chayne,8717,2521,5777,6189,5089,6957

102SBY,Sumren,Bergen,5012,1063,7937,9560,1115,5499

104SBK,Samira,Beckle,1140,9206,3898,8544,5937,8705

105NBT,Nellie,Bogart,3017,3342,5939,2479,3374,2297

106CGT,Cheryl,Grouth,9620,7160,5113,4803,5492 ,2195

107DGT,Danuta,Graunt,1583,7450,1026,7463,2390,6509

108JDN,Jaiden,Deckle,4064,4978,2984,3159,1464,4858

109JCK,Jimran,Caliks,6253,7962,2732,7504,2771,5193

110DDN,Deynar,Derran,6305,8817,5200,3647,3365,1256

LAB ASSIGNMENT 8

Write a program that will process a series of zero or more credit card transactions by reading in the traction record from a binary file transaction.dat, validating the credit card, and then outputting the verdict as to the validity of the credit card. A transaction record consist of the following fields stored in the list format:

Name on the credit card: first name, last name

Credit card number

Card expiry date: two digit month [space] two digit year

Transaction number: a positive integer up to 5 digit length

Date: day [space] month [space] year

Time: 24 hour time

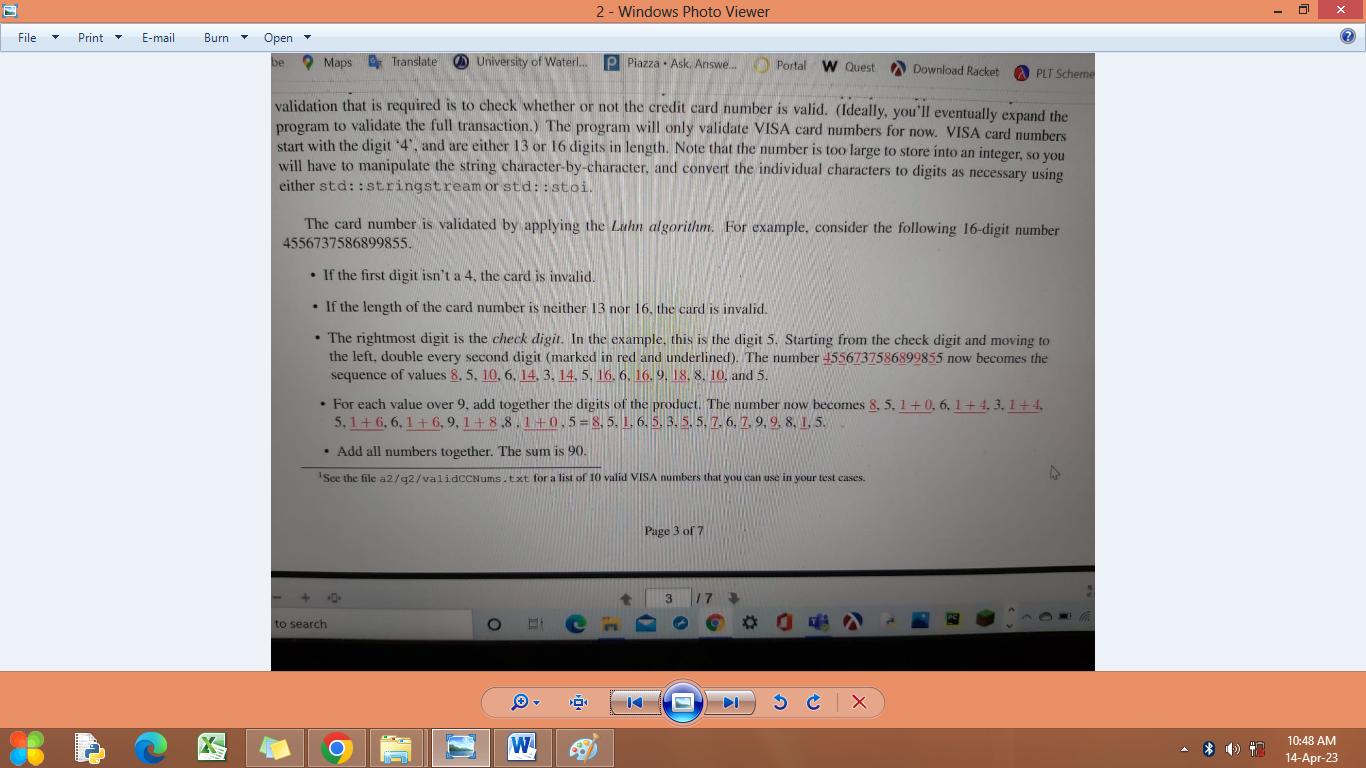
Amount:

Example: [Jane, Fred Smith, 4556737586899855, 05 23, 00001, 25 05 2021, 2105, 45.03]

You may assume that all of the information is correctly formatted and is of the appropriate type, so the only form of validation that is required to check whether or not the credit card number is valid. The program will validate VISA card numbers now. The card number is validated by applying Luhn algorithm. For example consider the following 16 digit number 4556737586899855.

* If the first digit is not a 4, the card is invalid
* If the length of the card number is neither 13 or 16, the card is invalid
* The right most digit is check digit. In the example, this is the digit 5. Starting from the check digit and moving to the left double every second digit ( marked in red and underline )



* Add all numbers together. The sum is 90
* Calculate the result of the sum modulo 10 i.e. 90 modulo 10 = 0
* If the result is 0 , the card number is valid

The format of each record output is as follows:

00001 25/05/2021 21:05 $45.03 4556737586899855 Jane 05/23 Valid

Note: Create a binary file transaction.dat with the following data

[Jane, Fred Smith, 4556737586899855, 05 23, 00001, 25 05 2021, 2105, 45.03]

[Frank, Streeter, 46724561782234, 06 25, 12002, 10 02 2020, 0910, 23.89]

[Heena, Shah, 4455667788992, 15 24, 00012, 11 09 2022, 1015, 48.45]

LAB ASSIGNMENT 9

Write a menu driven program using UDFs to perform the following binary file operations on Member.dat

* To create a binary file Member.dat / append Member data : Structure of member data is

{'MemberNo': 8003, 'Name': 'Archana',’Department’:HR’}

* Given a member no, display its associated name, else display appropriate error message.
* Remove all the members belonging to Sales Department from the file.
* Display all members

LAB ASSIGNMENT 10

Ria is a school librarian.

She wants a program to analyse pupil use of the library.

She wants to encourage reading by awarding gold, silver and bronze medals to the three pupils who have read the most books.

Test data has been stored in a binary file library.dat. Structure of each record in the binary file is (Student Index number, First name, last name, number of books read)

Write a program to read each record, calculate and display:

• the total number and average number of books pupils have read

• the IDs of pupils who have read fewer than ten books

• the details of the gold, silver and bronze medal winners.

Note: Create the binary file library.dat with the following details:

("105MS" , "Marcus" , "Smith" , 25 )

("103AZ" , "Anthony" , "Zarrent" , 5 )

("108MW" , "Matt" , "White" , 12 )

("112DB" , "Denise" , "Bilton" , 58 )

("124MK" , "Malcolm" , "Kelly" , 26 )

("116UK" , "Uzere" , "Kevill" , 29 )

("127AL" , "Abduraheim" , "Leahy" , 94 )

("124LS" , "Laura" , "Sampras" , 50 )

("121AP" , "Azra" , "Potter" , 61 )

("115AC" , "Anthony" , "Calik" , 10 )

("117PI" , "Pablo" , "Iilyas" , 49 )

("113MM" , "Mark" , "Montgomerie" , 68 )

("130FH" , "Felicity" , "Heath" , 11 )

("132JA" , "Jill" , "Alexander" , 61 )

["123SG" , "Sara" , "Grimstow" , 9 )

("134KD" , "Kevin" , "Dawson" , 74)

("122AB" , "Andrew" , "Bertwistle" , 42 )

("125JF" , "Jaide" , "Feehily" , 55 )

("128JS" , "Justin" , "Slater" , 68 )

("126CG" , "Colleen" , "Grohl" , 39 )

LAB ASSIGNMENT 11

You work for the Bank of Tawara. The bank sends a csv file encryptTransactions.csv containing recent transactions in an encrypted form. You will decrypt the  transactions . Each transaction must be decrypted and stored in a csv file decrypt.csv in the following format:

Following are the rules to decrypt the transactions:

 ·   the first digit and place this in the Transmitted check digit

• the next 6 digits and display these as the Bank sort code,

for example if the digits are 240912 the bank sort code would be displayed as 24-09-12

• the next 9 digits and display these as a 9-digit Account number

• the next 8 digits and display these in the Date digits

• the next character and display this in the Transaction

• any other digits and place these in the Working 2

·   Calculated check digit  to calculate a single digit checksum (by adding all the digits in the account number column and extracting the last digit from this total).

·   Working1  to compare the Transmitted and Calculated check digits.

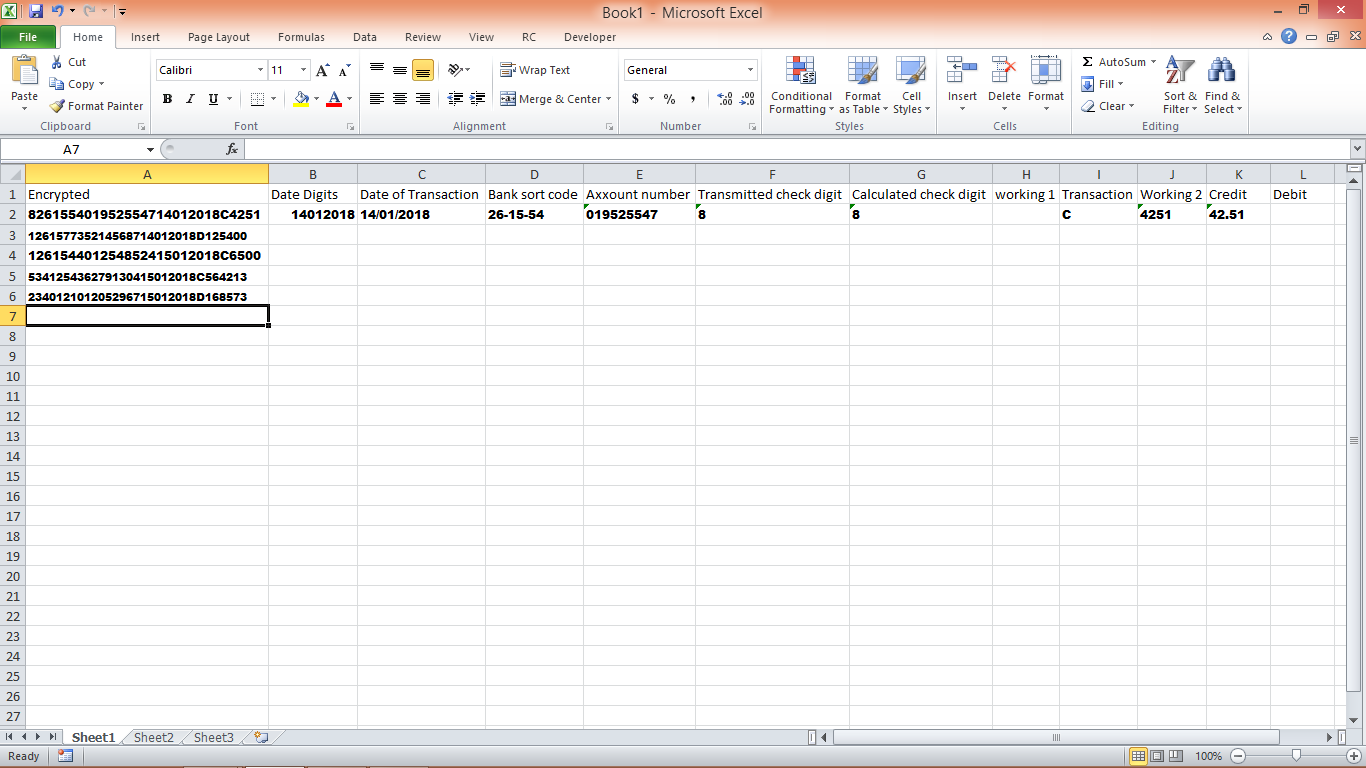
If the two data items for each transaction do not match, indicate that there is an error.

·   calculate the value of each transaction in Credit or Debit. The Transaction  must be used and the value held in the Working 2 is the value of each transaction in cents (there are 100 cents in a dollar). The letters C and D in the Transaction column represent Credit and Debit. either the credit or debit.

For example  if encryptTransactions.csv contain the following transactions:

|  |
| --- |
| 826155401952554714012018C4251 |
| 126157735214568714012018D125400 |
| 126154401254852415012018C6500 |
| 534125436279130415012018C564213 |
| 234012101205296715012018D168573 |

Then decrypt.csv must appear like this:



Decryption of the first transaction is given below:

Date digits: 14012018

Date of transaction: 14/01/2018

Bank sort code:  26-15-54

Account number: 019525547

Transmitted check digit: 8

Calculated check digit: 8

Working 1:

Transaction: C

Working 2: 4251

Credit: 42.51

Debit:

[Note : Create the csv file encryptTransactions.csv using Microsoft excel ]

LAB ASSIGNMENT 12

Write a menu driven program using UDFs to perform the following operations on a CSV file named PLANTS\_STOCK.csv

1. Add a new plant
2. Search a plant using its ReferenceID
3. Update a plant ( increase the price of all medium (‘M’) height plants by 1)
4. Display the plants whose Category ‘Climber’
5. Delete a plant using its ReferenceID

The structure of each row of the PLANTS\_STOCK.csv is as follows:

[Category,Name,Variety,Code,ReferenceID,Unit,Price,Height]

Sample data for PLANTS\_STOCK.csv is given below:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Category | Name | Variety | Code | ReferenceID | Unit | Price | Height |
| "Perennial" | "Aconitem" | "Monkshood" | "B" | "P24" | 1 | 3.00 | "M" |
| "Perennial" | "Anemone" | "Pink Star" | "P" | "P29" | 3 | 8.50 | "S" |
| "Perennial" | "Astilbe" | "Regal" | "WP" | "P32" | 1 | 3.50 | "M" |
| "Perennial" | "Echinops" | "Blue Wing" | "B" | "P38" | 1 | 3.25 | "M" |
| "Perennial" | "Hosta" | "Albo" | "V" | "P41" | 1 | 3.50 | "S" |
| "Perennial" | "Iris" | "Sibirica" | "Y" | "P42" | 3 | 6.00 | "M" |
| "Perennial" | "Phlox" | "Edward" | "RW" | "P46" | 1 | 3.30 | "S" |
| "Patio" | "Eunonymous" | "Mixed" | "VY" | "B07" | 12 | 12.50 | "S" |
| "Patio" | "Hebe" | "Varigatum" | "P" | "B11" | 1 | 12.00 | "M" |
| "Patio" | "Pieris" | "Carnival" | "B" | "B17" | 1 | 12.00 | "M" |
| "Patio" | "Begonia" | "Cocunut Ice" | "Y" | "B21" | 1 | 4.50 | "S" |
| "Patio" | "Begonia" | "Cocunut Ice" | "Y" | "B22" | 3 | 9.00 | "S" |
| "Patio" | "Begonia" | "Cocunut Ice" | "Y" | "B23" | 9 | 19.50 | "S" |
| "Patio" | "Dahlia" | "Moonfire" | "B" | "B24" | 1 | 5.50 | "M" |
| "Patio" | "Carnation" | "Trailing" | "RP" | "B35" | 5 | 8.25 | "M" |
| "Patio" | "Bamboo" | "Pleioblastus" | "N" | "B43" | 1 | 6.00 | "M" |
| "Climber" | "Clematis" | "Bouchard" | "P" | "C02" | 1 | 5.00 | "T" |
| "Climber" | "Rose" | "Compassion" | "Y" | "C11" | 1 | 8.00 | "T" |
| "Climber" | "Vine" | "Crimson Glory" | "P" | "C26" | 1 | 10.50 | "T" |
| "Climber" | "Honeysuckle" | "Belgica" | "YP" | "C51" | 1 | 8.25 | "M" |
| "Climber" | "Jasmin" | "Winter" | "YPB" | "C59" | 1 | 7.50 | "T" |
| "Climber" | "Wisteria" | "Wonder" | "B" | "C64" | 1 | 15.00 | "T" |
| "Fruit" | "Apple" | "Bramley" | "G" | "F17" | 1 | 16.50 | "T" |
| "Fruit" | "Apple" | "Cox" | "R" | "F19" | 1 | 16.30 | "T" |
| "Fruit" | "Apple" | "Discovery" | "G" | "F26" | 1 | 16.25 | "T" |
| "Fruit" | "Cherry" | "June" | "R" | "F31" | 1 | 16.00 | "T" |
| "Fruit" | "Pear" | "Conference" | "G" | "F32" | 1 | 16.00 | "T" |
| "Fruit" | "Plum" | "Victoria" | "P" | "F48" | 1 | 16.25 | "T" |
| "Fruit" | "Peach" | "Nancy" | "O" | "F72" | 1 | 16.00 | "M" |
| "Fuchsia" | "Annabel" | "Erne" | "W" | "U17" | 6 | 6.95 | "S" |
| "Fuchsia" | "Darkeyes" | "Bann" | "RW" | "U21" | 6 | 6.50 | "S" |
| "Fuchsia" | "Dancing Flame" | "Lagan" | "R" | "U25" | 6 | 6.95 | "M" |
| "Fuchsia" | "Thalia" | "Strule" | "PR" | "U37" | 6 | 6.50 | "S" |
| "Fuchsia" | "Swingtime" | "Shimna" | "PW" | "U52" | 6 | 6.25 | "M" |
| "Hedge" | "Berberis" | "Barberry" | "Y" | "H25" | 5 | 35.00 | "M" |
| "Hedge" | "Holly" | "Common" | "G" | "H29" | 1 | 50.00 | "T" |
| "Hedge" | "Bamboo" | "Blue Zion" | "B" | "H59" | 10 | 149.00 | "T" |
| "Hedge" | "Pyracantha" | "Orat" | "O" | "H60" | 5 | 30.00 | "M" |
| "Hedge" | "Dog Wood" | "Red Doa" | "R" | "H64" | 10 | 50.00 | "T" |
| "Shrub" | "Acer" | "Purple Mood" | "P" | "S08" | 1 | 18.00 | "T" |
| "Shrub" | "Buddleia" | "White Will" | "W" | "S14" | 1 | 7.50 | "M" |
| "Shrub" | "Hebe" | "Varigata" | "VR" | "S21" | 3 | 20.50 | "S" |
| "Shrub" | "Daphne" | "Purple Haze" | "P" | "S38" | 1 | 16.50 | "S" |
| "Shrub" | "Camellia" | "Sure Red" | "R" | "S60" | 1 | 13.00 | "M" |
| "Shrub" | "Hydrangea" | "Regal" | "P" | "S68" | 1 | 7.75 | "T" |
| "Shrub" | "Hydrangea" | "Elizabeth" | "B" | "S69" | 1 | 7.00 | "T" |
| "Shrub" | "Magnolia" | "Soulang" | "WP" | "S73" | 1 | 15.00 | "M" |
| "Shrub" | "Magnolia" | "Paul" | "W" | "S79" | 1 | 15.00 | "M" |
| "Shrub" | "Bay Tree" | "Standard" | "G" | "S81" | 1 | 70.00 | "M" |
| "Shrub" | "Bay Tree" | "Pyramid" | "G" | "S83" | 1 | 40.00 | "S" |

LAB ASSIGNMENT 13

A list contains following record of customer:

[customer\_name, Room\_Type]

Write the following user defined functions to perform given operations on the stack named Hotel.

1. Push\_Cust() – to Push customers names of those customers who are staying in ‘Delux’ Room\_Type.
2. Pop\_Cust() – To pop the names of customers from the stack and display them. Also display “Underflow” when there are no customers in the stack.
3. Display\_Status() – To display the content of the stack. Also display “Underflow” if the stack is empty.

For example:

If the list with customer details are as follows:

[“Siddarth”, “Delu”]

[“Rahul”,”Standard”]

[“Jerry”, “Delux”]

The stack should contain

Jerry

Siddarth

The output should be ( When Pop\_Cust() is called )

Jerry

Siddarth

Underflow

The output should be ( When Display\_Status() is called )

Jerry

Siddarth

Also write a menu driven program to call the user defined functions. The menu options are

1. PUSH
2. POP
3. DISPLAY
4. EXIT

Execution of the program must be terminated when the user selects the choice 4.EXIT

LAB ASSIGNMENT 14

A dictionary contains the details of vehicles – { car\_name:Maker}.

Write the following user defined functions to perform given operations on the stack named CAR:

1. PUSH(vehicle) where , vehicle is a dictionary containing details of vehicles – { car\_name:Maker}. The function should push the name of the car manufactured by TATA. (including all the possible cases like Tata, TaTa, etc) to the stack)
2. POP() – To pop the car on the top of the stack and display. If no car, display “Underflow”
3. COUNT() – To return number of cars in the stack. If no car, return ‘Underflow”.
4. DISPLA() – To display all the car names from the stack. If no car, display empty.

For example:

If the dictionary contains the following data:

{“Santro”:”Hyundai”,”Nexon”,”TATA”,”Safari”:”Tata”}

The stack should contain

Safari

Nexon

Also write a menu driven program to call the user defined functions. The menu options are

1. PUSH
2. POP
3. DISPLAY
4. COUNT
5. EXIT

Execution of the program must be terminated when the user selects the choice 5.EXIT

LAB ASSIGNMENT 15

A list contains following record of a customer:

[Customer\_name, Phone\_number, City]

Write the following user defined functions to perform given operations on the stack named ‘**status’**:

(i) Push\_element() - To Push an object containing name and Phone number of customers who live in Goa to the stack

(ii) Pop\_element() - To Pop the objects from the stack and display them. Also, display “Stack Empty” when there are no elements in the stack.

For example:

If the lists of customer details are:

[“Gurdas”, “99999999999”,”Goa”]

[“Julee”, “8888888888”,”Mumbai”]

[“Murugan”,”77777777777”,”Cochin”]

[“Ashmit”, “1010101010”,”Goa”]

The stack should contain

[“Ashmit”,”1010101010”]

[“Gurdas”,”9999999999”]

The output should be:

[“Ashmit”,”1010101010”]

[“Gurdas”,”9999999999”]

Stack Empty

Also write a menu driven program to call the user defined functions. The menu options are

1. PUSH
2. POP
3. EXIT

Execution of the program must be terminated when the user selects the choice 3.EXIT

**LAB ASSIGNMENT 16**

Create a table ORDER, without any constraints based on the data given below.



1. Write an SQL statement to find the total purchase amount for all orders.
2. Write a query that counts the number of salesmen with their order date for each day.
3. Write an SQL statement that counts all orders for a date October 5th, 2012.
4. Write an SQL statement to find the highest purchase amount with their ID, for only those

salesmen whose ID is within the range 5003 and 5008.

1. Write an SQL statement to change the salesman\_id to a number incremented by 1000 for all salesmen(5002 should be 6002, 5005 should be 6005 and so on)
2. Write an SQL statement to display the Purchase amount increased by 3% for those salesmen who have more than 1 order. (Use column Alias)
3. Write the command to add a field SalesmanName with appropriate data type.
4. Write the command to set ord\_no as Primary key.
5. Write the command to display the Total Purchase Amount generated by each salesman.
6. Write the command to display the number of orders and the minimum Purchase Amount in the month of September.

**LAB ASSIGNMENT 17**

Consider the tables given below and write SQL commands as per the requirement





1. To display names of all staff who are in Sales department having experience less than 9 years and commission more than 700.
2. Display the average salary of staff working in Finance department

(Salary= Basic+Allowance)

1. Update the Basic and Allowance of Aryan to 13000 and 55000 respectively.
2. Display the number of staff with a Basic Salary more than 30000, along with their Department.
3. Display the number of departments with more than 1 male staff.
4. Display the minimum salary and maximum commission earned by any staff in the company.
5. Display the number of departments with only female teachers in Sales Department

EX: 18

Golden key Hospitals owns hospitals around the world. Robin Sharma is the owner of the company.

Robin has collected information about the hospitals and their customers. The **PATIENT** table contains details of customers from a region in England.

The structure of the **PATIENT** table is:



**GENDER**

Create the PATIENT table with the above details.

Write a python connectivity program to perform the following tasks.

1. Insert 5 records into the PATIENT Table

Sample record:



**GENDER**

1. Sort AND DISPLAY the **PATIENT** table into alphabetical order by **PATIENT NAME**
2. Robin wants a list of Patients who are females. He wants all fields displayed. Create a query on the **PATIENT** table to find this information.

4)Robin wants a list of patients whose Date of birth is between 1 April 1987 and 30 April 1987.

**LAB ASSIGNMENT 19**

Simon is the owner of *Celtic Springs Scuba (CS Scuba).* The company teaches diving and

offers diving trips around the world. Simon has collected information about the holidays. He wants to put the holidays information in a table DESTINATIONS.

The structure of the **DESTINATIONS** table is:



**Create the DESTINATION table using the above details and insert five records.**

Simon wants a MySql python connectivity program to perform the following tasks:

1. Add these details to the **DESTINATIONS** table.

****

1. Display the records of the **DESTINATIONS** table sort by **COUNTRY** into alphabetical order.
2. Simon wants a list of holidays that:

• are liveaboard

• are 7 days duration

• use Blue Water dive boats.

The list must:

• show only these fields in this order **Holiday ID, Region, Country**

• be sorted in descending order of **Holiday ID**

**LAB ASSIGNMENT 20**

**ABC infotech Pvt.Ltd. needs to store, retrieve and delete the records of its employees, Develop an interface that provides front-end interaction through python , and stores and updates records using MYSQL.**

**The operations on MYSQL table “EMP” involving Adding, reading, searching, updating and deleting records of employees as per user’s request.**

**Following is the structure of EMP table:**

|  |  |
| --- | --- |
| **Field Name** | **Description** |
| **Title** | **Mr or Mrs or Miss** |
| **EmpNo** | **Employee Number** |
| **FName** | **First Name** |
| **LName** | **Last Name** |
| **Dept** | **Department Name** |
| **Sal** | **Salary** |
| **DOJ** | **Date of Join** |
| **Gender** | **M or F** |
| **Desig** | **Designation** |