Order of pages

1. Folder provided by the college
2. Practical index sheet
3. Cover page
4. Lab Sheet title page (Lab-sheet #1)
5. Acknowledgement
6. Objectives (write a paragraph)
7. Table of contents with page number
    7.1 Theory (write theory in short form)
    7.2 Coding (write the code here)
    7.3 sample Output

(Insert the screenshot; Make sure that it contains your local computer name with folder (A complete path of program))

8. Discussion and conclusion
9. Bibliography

Note:

1. Capitalize each word. / Be clear, informative & concise. / Avoid abbreviations,
2. Spacing------------> Double-space
3. Page Numbering-> Bottom of the page, center
5. Headings----------> Bold font on a separate line / No numbering for headings & subheadings
6. Paper--------------> A4 size, white, one-sided writing
7.Page margins----> Left 1.25”, Right 1”, Top 1”, Bottom 1”
8. Page Binding----> Spiral or hard stick
9. A4 size paper; white in color
10. Write on one side/front only; the back side must be blank.

You must write manually; computerized reports will not be accepted!

As you finish lab report one, repeat the same process for other reports.

files.fm/u/84qau5qpn8

Lab Work - 1 (Function)

1. WAP to know a number is even or odd using function named evenodd().
2. WAP to print the greatest value among three numbers using a function int great().
   We have to use return statement.
3. WAP to know a number is prime or composite using function void prime_composite(int).
4. WAP to find sum of series 1, 2, 3,…..200 using function.
5. Assume yourself function WAP to find sum of two numbers using function named sum().
   name. It returns an integer value.
6. WAP to input elements of an array and print them with their sum. Suppose, the array is one dimensional and is of void type and function to be used is array_elements().
7. Suppose a function void matrix_sum(int a[][], int b[][]).Here, we have passed array as parameter. Use this function to find sum of matrices.
8. WAP to sort 'n' number of strings using function. Pass strings as parameter.
9. WAP to find factorial value of a number using recursive function int factorial(int).

Lab Work - 2 (Structure and Union)

1. WAP to input id, name and grade for 10 students. Then print them. Use array of structure concept.
2. WAP to input 20 employees name, designation and salary. Then search a record of an employee and its position on the basis of name.
3. We have two structures as given below.

   ```
   struct employee
   {
      char ename[100];
      char eaddress[100];
   };
   struct salarydetails
   {
   char pos[100];
   float salary;
   struct employee detail;
   }payinfo;
   ```

   WAP to input 5 employees' details with salary then print them. Use nested struct concept.

4. WAP to input id, name and address of 20 students using struct. Then print them in sorted format on the basis of name.
5. WAP to input any 10 teacher's id, name and subject using 'typedef' and structure. Then print them on screen.
6. WAP using function and structure to calculate sum of two distances measured in terms of kilometers and meters. For example, If we input following data

Kilometer	meter
23 (km1)	445(m1)
45 (km2)	756(m2)
69	201

   The output would be 69 km and 201 meters.
7. WAP to input student id, name and grade and print them. Use union concept.

files.fm/u/dshgxm9t3f

Lab Work - 3 (Pointer)

1. WAP to perform arithmetic calculations (sum, difference, multiplication and division) of two numbers using pointers.
2. WAP to know a number is even or odd using pointer.
3. WAP to find sum and average of ‘n’ natural numbers using pointer.
4. Use array to input 10 elements and print them. Use array as pointer.
5. WAP to input 10 elements and print maximum and minimum value. Use array as pointer.
6. WAP to swap two values using call by reference and call by value.
7. WAP to sort 10 numbers stored in an array using pointer.
8. WAP to print multiplication table of a number using pointer.

Lab Work - 4 (Data File)

1. WAP to input a sentence and store data in a data file “file.txt”. And print them on screen. Use getc() and putc() functions.

2. WAP to store some ‘n’ natural numbers in a data file and print them on screen. Use getw() function. Also print their average.

3. WAP to store student’s name and address in a data file “student.txt”. Use fprintf() function. Then read contents of data file and print them on screen. Use fscanf() function.

4. WAP to store book’s name, edition and price in a data file “book.txt” using yes/no options. It means the computer stores data until you say ‘n’. Then print them on screen.

5. WAP to store employee’s name, designation and salary in a data file “employee.txt” using fwrite(). The computer stores data until you say ‘n’. Then read the contents of that file and print on screen using fread().

6. WAP to input students’ name, grade and marks in five subjects. Then store these all data with total and percentage in a data file “student.dat”. Print all those data of students who have percentage >=80. You may use fprintf() or fwrite().

7. WAP to show the concept of rename() and remove() functions.

8. WAP to show concept of ftell(), fseek() and rewind() functions.

files.fm/u/eu5wgfw7wd

Lab Work - SQL

A. Database practical session-1

a) Create a database named ‘EMPLOYEE’.

b) Get inside the database.

c) Create a table named ‘employee’ with following fields.
Employee_id—int primary key not null auto increment
Employee_name--→varchar(100)

d. Delete the table.

e. Know whether that deleted table exists or not.

f. Delete the database which you have created (employee).

g. Know whether that deleted database exists or not.

files.fm/u/eu5wgfw7wd

B. Database practical session-2

a. Create a database named ‘student’.

b. Create a table ‘student’ with following fields.

  Student_id-------------→integer primary key   auto increment
  Student_name---------→varchar(100)
  Student_address-------→varchar(100)
  Student_grade----------→integer(int)

c. Insert any 6 records using ‘insert’ and ‘values’ command.

d. Display all the records of all fields.(Use select *)

e. Display all the records of fields’ student_id and student_name.

f. Display records of students whose name starts with letter ‘a’.

g. Display records of students whose name ends at letter ‘y’.

h. Display all the records of students whose grade is 12. Use ‘where’ command.

i. Display all the records of students whose grade is 12 and who are from address “Kathmandu”. Use ‘and’ operator.

j. Update the student name with any other name who has id 1.

k. Delete the record of student whose id is 3.

l. Display the records of students in sorted order using field ‘student_name’.
Use ‘order by field name asc/desc’.

m. Alter the table with following fields.

  Add one more field student_section---→varchar(100). [Use alter and add command]*-*-

  Change the size of field student_address--→200 [Use alter and modify command] filebin.net/vogr7n5wkumvqib8

C.Database practical session-3

a. Create a database named '[your-section]_employees'.
b. Create a table 'employee' with following fields.
   emp_id-------------→integer primary key   auto increment
   emp_name---------→varchar(100)
   emp_position-------→varchar(100)
   emp_salary----------→float
c. Insert any 6 records using 'insert' and 'values' command.
d. Display all the records of all fields.(Use select *)
e. Display all the records of fields' emp_id and emp_name.
f. Display records of employees whose name starts with letter 'ab'.
g. Display records of employees whose name ends at letter 'y' and starts from 'b'.
h. Display all the records of employees whose position is 'engineer'.
i. Display all the records of employees whose salary is in range 30000-45000.
j. Update the employees name with any other name who has id 1.
k. Delete the record of employees whose id is 3.
l. Display the records of employees in sorted order. Use 'order by field name asc/desc'.
m. Alter the table with following fields.
   Add one more field employees_address ---→varchar(100). [insert after name field]
   Change the size of field employees _position--→200 [Use alter and modify command]
n. Print the record of employee who has the maximum and minimum salary in database.
o. Print the records whose salary distribution is more /less than average in database.

[Note: Use both shell interface and graphical interface for these operations]

Lab Work - JS(Javascript)

1. Write a JS program to find area and circumference of circle.
2. Write a JS program to find the square root of a number. Here the number is entered by the user.[Use Math object with method (Math.sqrt for square root) and prompt method to input data]
3. Write a program to pick the greatest number among three numbers entered by the user.
4. Write a program to input a number and know that it is an even or odd number. [Use prompt for input and % operator to get remainder]
5. Write a program to input your name and age using form and know whether he/she is eligible to vote or not. [Use function and onclick event]
6. Write a program to perform following operations using switch structure.
   If you enter A+, it says you have obtained grade A+.
   If you enter A, it says you have obtained grade A.
   If you enter B+, it says you have obtained grade B+.
   If you enter B, it says you have obtained grade B.
   If you enter any other value/character, it says you have obtained other grade.
7. WAP to print factorial value of a number.
8. WAP to print multiplication table of a number. Use function.
9. Write a program to reverse a string. [use loop from its length-1 to 0 and print it]
10. An array contains multiple strings. Print them in reverse order.[use array.reverse() method].mk-
11. Write a program to insert a string “We are learning JS” via form object getElementById() method. Use paragraph with id.
12. Input your name and grade using form. Validate this with a message ‘This field is required’, using onclick() event.
13. Write any two paragraphs in your html page. Now write a program using JQuery to hide that written paragraphs as you click on that.
14. Use JQuery to create an alert message for a button.

Lab Work – PHP

1. WAP to find sum of two numbers. Here, numbers are entered via html form.
2. Write a PHP program to swap two numbers.
3. Write a PHP program to input name, address using form, and display them.
4. Write a PHP program to find the greatest number among three numbers. Here, numbers are entered via html form.
5. Write a PHP program to input any number via form and check whether the number is positive or negative and be sure number should not be zero.
6. Write a PHP program to display your name 100 times.
7. Write a PHP program to print a multiplication table of a number. Input the number via form.
8. Write a PHP program to display the factorial value of a number.
9. WAP to print factors of a number.
10. Write a PHP program to store some strings in an array and print them.
11. Write a PHP program to sort data stored in an array. [Use sort() function and print_r() to print.]
12. Write a PHP program to know whether two strings are the same or not.
13. Write a PHP program to input a string through a text box and find its length.[use strlen()]
14. Write a PHP program to connect a database named ‘student’.
15. Write a PHP program to insert records in a database named student having table student with field’s id, name and grade.
16. Write a PHP program to display all records (previous question) of students who are in grade 11.
17. Write a PHP program to delete records of a student whose id is 3.[previous question]
18. Write a PHP program to update a record of student whose id is 2. [Try same program with name/grade.]

Case study /project work in PHP:-

To make a website with CRUD operations.
Kindly open the pdf file to know the steps.