Practical 1

330en30

Create your e-mail account on Gmail/Yahoo mail/Hotmail.

Objective

After completion of this practical you will be able to create your personal e-mail account.

Software required

- Internet Explorer / Mozila /Google Chrome (any browser).

Pre-requisite

- You should be able to use Computer/Laptop.
- You should have the knowledge of internet.

Procedure

- Make sure your computer is "On" and Internet service is enabled there.
- Open any web browser (i.e., Internet Explorer, Mozila, Google Chrome, etc.).
- Type URL *www.gmail.com* (for gmail account).
- Click on sign up button.
- Fill in the registration form.
- Read the terms and conditions then accept it.
- Submit the filled form.
- Now your account is created and you can use it for communication.

Dos and Don'ts

- Choose username that could relate to your name.
- Do not use very simple password.

Notes

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 2

XYZ enterprises, a pharmaceutical company is heading for its annual general body meeting. You have been assigned the work to create a document using OpenOffice Writer highlighting the important achievements of your company. Also form a spreadsheet in Calc showing the item wise sales in each month. The spreadsheet should also show the average sales figures and the total annual sales. Finally, form a 3 minutes presentation using Impress highlighting your company, its objective, estimated sales target and the achieved sales in the last financial year. (Note : You may assume the sales figures)

Objective

After completion of this practical you will be able to use Office productivity tools like Writer, Calc and Impress.

Software required

- OpenOffice Writer
- OpenOffice Calc
- OpenOffice Impress

Pre-requisite

- Theoretical knowledge of text documents, spreadsheets and presentations.

Procedure

- Open OpenOffice Writer.
- Form a document that showcases the company and its objectives. Also write the major achievements for the last financial year.
- Format the document using appropriate font size, style and colour.
- Save your file by any name (say ABCdoc.odt).
- Open OpenOffice Calc.

- Form a spreadsheet showing the sales figures of various items produced by your company. Take minimum five such items i.e., names of the medicines in this case.
- Use appropriate functions to display average sales and total sales.
- Save your file with an appropriate name.
- Open OpenOffice Impress.
- Create a presentation by inserting minimum 10-15 slides showing the company's progress and sales.
- Insert appropriate animations and transitions in the presentation.
- Save the document by any name.

Dos and Don'ts

- Remember the file name and the location where you are saving the file.

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 3

Write a C++ program to accept two numbers, a float and an integer and display the following :-

a) Sum of two numbers in integer form.

b) Sum of two numbers in float form.

Objective

After completion of this practical you will be able to use basic data types, operators and type conversion in C++.

Software required

- Turbo C++

Pre-requisite

Learner should have the basic knowledge of characters, tokens, basic data types, input output streams and type conversion.

Procedure

- Accept two numbers using input stream.
- Add two numbers using arithmetic operators.
- Display the numbers in integer and float form using output stream.

Dos and Don'ts

- Add your name and enrolment number as a comment on the top of the program.
- Save the program with proper filename having .cpp extension.
- Create a folder and store all the programs in the folder.

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 4

Write a C++ program that accepts radius and then displays the area and perimeter of a circle. Consider “Pi” as a constant with value 3.14.

Objective

After completion of this practical you will be able to use the basic data types, operators, constants and variables in C++.

Software required

- Turbo C++

Pre-requisite

Learner should have the basic knowledge of characters, tokens, basic data types, constants and variables and input output streams.

Procedure

- Accept two numbers using input stream.
- Calculate area of the circle (pi*r*r) and perimeter (2*pi*r).
- Display the result in appropriate form using output stream.

Dos and Don’ts

- Add your name and enrolment number as a comment on the top of the program.
- Save the program with proper filename having .cpp extension.
- Create a folder and store all your programs in the folder.

Learner’s Observations

....

....

....

Notes

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 5

Write a C++ program to accept a number. Using conditional operator print whether the number is even or odd.

Objective

After completion of this practical you will be able to use conditional operator.

Software required

- Turbo C++

Pre-requisite

- Learner should have the basic knowledge of C++ program.

Procedure

- Accept a number from the user using input stream.
- Use conditional operator for finding whether the number is even or odd.
- Syntax of conditional operator

  (**condition)?statement if condition is true: statement if condition is false**

- Display the result.

Dos and Don’ts

- Add your name and enrolment number as a comment on the top of the program.
- Save the program with proper filename having .cpp extension.
- Create a folder and store all the programs in the folder.

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 6

Write a C++ program to accept a character. Print whether the character is an alphabet, digit, or a special character. Display appropriate messages.

Objective

After completion of this practical you will be able to use conditional statements like:-

if ….. else

if …… elseif…else

Software required

- Turbo C++

Pre-requisite

- Learner should have the basic knowledge of writing a C++ program and also know about compound, null and conditional statements.

Procedure

- Accept a character from the user.
- Use if … else statement for finding the character is alphabet, digit, or a special character.
- Syntax of if statement

```
    if(condition)
            { code when true }
    else
            { code when not true }
```

- Display the result.

Dos and Don’ts

- Add your name and enrolment number as a comment on the top of the program.
- Save the program with proper filename having .cpp extension.
- Create a folder and store all the programs in the folder.

Learner’s Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor’s signature

Practical 7

Write a C++ program to accept a number and display the corresponding number of week day

E.g., If the user enters “1” display “ Monday”, if the user enters “2” display “ Tuesday”…… and so on.

Objective

After completion of this practical you will be able to use conditional statement, switch…… case.

Software required

- Turbo C++

Pre-requisite

- Learner should have the basic knowledge of a C++ program and know about switch … case construct.

Procedure

- Accept a number from the input stream.
- Use switch case statement like

  Syntax

```
switch (variable)
 {
 case 1: statement1 ; break;
 case 2: statement 2 ; break;
 case 3:statement 3;  break;
 .
 .
 .
 default: default statement break;
}
```

- Display the result.

Dos and Don’ts

- Add your name and enrolment number as a comment on the top of the program.
- Save the program with proper filename having .cpp extension.
- Create a folder and store all the programs in the folder.
- Use your own logic building skills to develop the program.
- Display appropriate messages.

Learner’s Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor’s signature

Practical 8

Write a C++ program (using while loop) to display all the multiples of 5 from 100 to 50.

Objective

After completion of this practical you will be able to use looping statement while ( ).

Software required

- Turbo C++

Pre-requisite

- Learner should have the basic knowledge of writing a C++ program and know the working of while loop.

Procedure

- Use while loop
- Syntax

```
while(test-expression)
{
        body of the loop
        Increment / decrement
}
```

- Display the result.

Dos and Don’ts

- Add your name and enrollment number as a comment on the top of the program.
- Save the program with proper filename with .cpp extension.

- Create a folder and store all the programs in the folder.
- Use your own logic building skills to develop the program.

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 9

Write a program using do..while loop to display the Fibonacci series 0, 1, 1, 2, 3, 5… n

Objective

After completion of this practical you will be able to use do …… while statement.

Software required

- Turbo C++

Pre-requisite

- Learner should have the basic knowledge of working of conditional and looping statements.

Procedure

- Use do.. while loop in the program.
- Use logic for displaying Fibonacci series 0,1,1,2,3,5,8….
- Syntax

```
do
{
        Body of the loop
        Increment / decrement statements
}while(test-expression);
```

- Display fibonacci series.

Dos and Don'ts

- Add your name and enrolment number as a comment on the top of the program.
- Save the program with proper filename having .cpp extension.

- Create a folder and store all the programs in the folder.
- Use your own logic building skills to develop the program.

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 10

Write a C++ program using for loop to find whether the number is an Armstrong number or not.

Objective

After completion of this practical you will be able to use “for loop”.

Software required

- Turbo C++

Pre-requisite

- Learner should have the basic knowledge of conditional and looping statements.

Procedure

- Accept a number from the input stream.
- Use for loop in the program.
- Syntax

```
for (initialization; condition; increment/decrement)
{
        Body of the loop
}
```

- An Armstrong number is one whose sum of cube of digits is the number itself. For example 153 is an Armstrong number.

  153=1^3+5^3+3^3

- Use the required logic in the program for finding if the given number is Armstrong or not.
- Display the relevant result.

Dos and Don'ts

- Add your name and enrolment number as a comment on the top of the program.
- Save the program with proper filename having .cpp extension.
- Create a folder and store all the programs in the folder.
- Use your own logic building skills to develop the program.

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 11

Write a menu driven program to swap two numbers using call by value and call by reference.

Objective

After completion of this practical you will be able to structure your program where a large program can be divided into smaller, simpler modules using user defined functions.

Software required

- Turbo C++

Pre-requisite

- A basic knowledge of functions (prototype, definition, call with arguments and return types).

Procedure

- Accept two numbers from the input stream.
- Create function for swapping two numbers.
- Use call by value method and swap two numbers.
- Use call by reference method and swap two numbers.

Syntax for function prototype:

```
return-type function_name( arg-types);
```

Syntax of function definition

```
return-type function_name( arg-types)
{
declarations;
statements;
}
```

Dos and Don’ts

- Add your name and enrolment number as a comment on the top of the program.
- Save the program with proper filename having .cpp extension.
- Create a folder and store all the programs in the folder.
- Use your own logic building skills to develop the program.

Learner’s Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor’s signature

Practical 12

Create three functions by the name CALCULATE() according to the prototypes given below:

- void CALCULATE(int) // to check whether the argument being passed is even or odd
- int CALCULATE(int, int) // to return the sum of two numbers being passed as arguments.
- void CALCULATE() // to display the message " YOU ARE DOING GREAT"

Objective

After completion of this practical you will be able to structure program where a large program can be divided into a smaller, simpler modules using user defined functions.

Software required

- Turbo C++

Pre-requisite

- A basic knowledge of function prototype and function declaration is required.

Procedure

- Create function CALCULATE.
- Syntax for function prototype:

```
return-type function_name( arg-types);
```

- Syntax of function definition

```
return-type function_name( arg-types)
{
```

Notes

```
            declarations;
            statements;
      }
```

- The functions with same function name are distinguished by the number and type of arguments (Function signature).

Dos and Don’ts

- Add your name and enrolment number as a comment on the top of the program.
- Save the program with proper filename having .cpp extension.
- Create a folder and store all the programs in the folder.
- Use your own logic building skills to develop the program.

Learner’s Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor’s signature

Practical 13

Write a C++ program to accept two strings and do the following:

1. Compare the two Strings
2. Display the length of the strings
3. Append the strings.

Objective

After completion of this practical you will be able to use header files and built in functions.

Software required

- Turbo C++

Pre-requisite

- A basic knowledge of header files and built in functions is required.

Procedure

- Accept two strings from the input stream.
- Use the following string functions

    1) strcmp(S1,S2) - to compare two strings

    2) strlen(S1) - to find the length of the string

    3) strcat(S1,S2) - to append one string after another

    (S1 and S2 are string variables)

- Display the result.

Dos and Don'ts

- Add your name and enrolment number as a comment on the top of the program.
- Save the program with proper filename having .cpp extension.

- Create a folder and store all the programs in the folder.
- Use your own logic building skills to develop the program.

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 14

Write a C++ program to accept ten numbers using an array. Search for a number in an array using binary search method.

Objective

After completion of this practical you will be able to create, initialize and traverse a one dimensional array using C++ and also understand the logic of binary search.

Software required

- Turbo C++

Pre-requisite

- Basic knowledge of arrays and their working is required.

Procedure

- Accept 10 numbers in an array.
- Accept a number, say S.
- Check whether the number S exists in the array using binary search method.
- If number exists display “ NUMBER EXITS in .... LOCATION” else “NUMBER NOT FOUND**”.**

Dos and Don’ts

- Add your name and enrolment number as a comment on the top of the program.
- Save the program with proper filename having .cpp extension.
- Create a folder and store all the programs in the folder.
- Use your own logic building skills to develop the program.

Notes

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 15

Write a C++ program to accept eight numbers in an array. Sort the numbers in ascending order using Bubble sort method. Also insert a number in the sorted array.

Objective

After completion of this practical you will be able to sort array in ascending or descending order.

Software required

- Turbo C++

Pre-requisite

- Basic knowledge of arrays and its elements, their working and sorting technique logic (Bubble Sort) is required.

Procedure

- Accept 8 numbers in an array.
- Sort the numbers in ascending order using Bubble sort method.
- Print the sorted list.
- Accept a new number.
- Insert the number in the appropriate position in the sorted list.
- Display all the nine numbers of the array.

Dos and Don'ts

- Add your name and enrolment number as a comment on the top of the program.
- Save the program with proper filename having .cpp extension.
- Create a folder and store all the programs in the folder.
- Use your own logic building skills to develop the program.

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 16

Write a C++ program to accept two arrays and merge them to form a new array such that the new array contains the elements from both the arrays alternatively.

Objective

After completion of this practical you will be able to merge two or more arrays according to the given conditions.

Software required

- Turbo C++

Pre-requisite

- Basic knowledge of working with arrays and assigning the values of one array to another is required.

Procedure

- To accept the elements of two arrays, say A and B of any size.
- Create a new array C such that the elements of array A and array B are alternatively placed in array C.

  For example

  Array A contains 1,2,3,4,5,6

  Array B contains 7,8,9,10,11,12

  Then array C should contain 1,7,2,8,3,9,4,10,5,11,6,12

Dos and Don'ts

- Add your name and enrolment number as a comment on the top of the program.
- Save the program with proper filename having .cpp extension.
- Create a folder and store all the programs in the folder.
- Use your own logic building skills to develop the program.

Notes

Learner’s Observations

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

Instructor’s signature

Practical 17

Write a C++ program to accept 4 X 4 matrix and find the sum of odd numbers of the matrix.

Objective

After completion of this practical you will be able to create, traverse and manipulate the elements of a 2D array.

Software required

- Turbo C++

Pre-requisite

- Basic knowledge of 2 D arrays and its working is required.

Procedure

- Accept the elements of a 4×4 matrix using nested for loops.
- Check each element of the matrix, whether it is odd or even.
- If the number is odd then add the value to a variable called total.
- Display the matrix.
- Display the sum of odd elements of the matrix.

Dos and Don’ts

- Add your name and enrolment number as a comment on the top of the program.
- Save the program with proper filename having .cpp extension.
- Create a folder and store all the programs in the folder.
- Use your own logic building skills to develop the program.
- Try to use mininum number of loops.

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 18

Create a structure named General_Date with day(int), month(string) and year(int) as its members. Create another structure named Employment_Exchange with application_no(int), Application_name(String), DOB(instance of General_Date), Qualification(String) and Family_income(float).

Write a C++ program to accept the members of the structure Employment_Exchange. If the family income is less than 1000 then display the message "Eligible for Rozgar Yojna" else display for " Not Eligible for Rozgar Yojna".

Objective

After completion of this practical you will be able to use user defined data type structures.

Software required

- Turbo C++

Pre-requisite

- Knowledge of creating and accessing the members of the structure is required.

Procedure

- Create a structure prototype as follows:

```
struct Employment_Exchange
{
    int application_no;
    ……
    …….
};
```

- Creating the instance of a structure

  Structure name variable name;

  For example,

  Employment_Exchange E1;

- Accessing of data members (using dot operator)

  Structure_variablename.member_name ;

  For example,

  E1.application_no=10;

Dos and don'ts

- Add your name and enrolment number as a comment on the top of the program
- Save the program with proper filename having .cpp extension
- Create a folder and store all the programs in the folder.
- Use your own logic building skills to develop the program.

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 19

Create a class named competition with the following members:-

Private members

Member Name	Data Type
learner_id	integer
learner_name	string
category	char (junior/senior)
competition_name	string
position	integer
points	integer

Create a function calculate ( )- to assign the points according to the value of position given below:

Position	Point
1	10
2	7
3	5
>3	0

Public members

1. Create an appropriate function Accept() and Display() to accept and display data respectively.

2. A constructor to initialize all the numerical members to 0 and all strings to “WELCOME”.

   Write a C++ program to implement the class defined above by creating an object.

Objective

After completion of this practical you will be able to use user defined data types in OOP.

Software required

- Turbo C++

Pre-requisite

- Basic knowledge of OOP, defining classes, declaring objects and accessing public members is required.

Procedure

- Create a class called Competition containing public and private members as given in the problem statement.
- Use the following syntax.

```
class classname
{
    private:
    variable declaration;
    function declaration;
    public:
    variable declaration;
    function declaration;
};
```

Dos and Don'ts

- Add your name and enrolment number as a comment on the top of the program.
- Save the program with proper filename having .cpp extension.
- Create a folder and store all the programs in the folder.
- Use your own logic building skills to develop the program.

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Notes

Practical 20

Create a class called COURSE with the following members:

Members	**Data type**
Private members:	
course_id	integer
course_name	string
duration	integer
fee	float
Public memebrs:	
void accept()	

Create a class called SR_SECONDARY which is **publically inherited** from the class COURSE and has the following members

Members	**Data type**
Private members:	
subject1	string
subject2	string
subject3	string
Public function:	
void enter()	
void display()	

Define a class called VOCATIONAL that is **protectedly derived** from class SR_SECONDARY. The members of the class are as follows :

Members	Data type
Private members:	
voc_name	string
requirement	string
age	integer
Public function:	
void voc_accept()	
void voc_display()	

Implement the above classes to accept and display data of Sr_Secondary class and vocational class.

Objective

After completion of this practical you will be able to use multilevel inheritance.

Software required

- Turbo C++

Pre-requisite

- Basic knowledge of inheritance is required. Learners should know about different types of inheritance (simple, multiple and multilevel) and visibility modes (public, private and protected).

Procedure

- Create three classes called COURSE, SR_SECONDARY, VOCATIONAL.
- Give appropriate visibility modes for the members of each class and also give appropriate type of inheritance.
- Create objects to display data of Sr_Secondary class and vocational class.

Dos and Don’ts

- Add your name and enrolment number as a comment on the top of the program.
- Save the program with proper filename having .cpp extension.

- Create a folder and store all the programs in the folder.
- Use your own logic building skills to develop the program.

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 21

Create a class called FOUR_WHEELER with the following members:-

Members	**Data type**
Private members:	
model_no	integer
colour	string
price	float
engine_cc	float
Public function:	
void accept()	
void display()	

Create a class called TWO_WHEELER with the following members:-

Members	**Data type**
Private members:	
type	string
HP	integer
Public function:	
void two_accept()	
void two_display()	

Create a class called MOTOR_VEHICLE that is derived in protected mode protectedly from FOUR_WHEELER and in public mode TWO_WHEELER. The members of the class MOTOR_VEHICLE are as follows :

Members	**Data type**
Private members:	
HP	string
protected members:	
category	string
Public function:	
void motor_accept()	
void motor_display()	

Implement the above class hierarchy.

Objective

After completion of this practical you will be able to use multiple inheritance.

Software required

- Turbo C++

Pre-requisite

- Basic knowledge of inheritance is required. Learners should know about different types of inheritance (simple, multiple and multilevel) and visibility modes (public, private and protected).

Procedure

- Create three classes called TWO_WHEELER, FOUR_WHEELER, MOTOR_VEHICLE.
- Give appropriate visibility modes for the members of each class and also give appropriate type of inheritance.
- Create objects to implement the class hierarchy.

Dos and Don’ts

- Add your name and enrolment number as a comment on the top of the program

- Save the program with proper filename having .cpp extension.
- Create a folder and store all the programs in the folder.
- Use your own logic building skills to develop the program

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Notes

Practical 22

Write a C++ program to reverse a number using pointers.

Objective

After completion of this practical you will be able to use pointers.

Software required

- Turbo C++

Pre-requisite

- Basic knowledge of pointers is required.

Procedure

- Accept a number.
- Create a pointer using * operator and assign the address using & operator.
- Use a pointer to access the number.
- Use % and / operators to separate the digits.
- Display the reverse number

  For example, if the number is 1234, output should be 4321

Dos and Don'ts

- Add your name and enrolment number as a comment on the top of the program.
- Save the program with proper filename having .cpp extension.
- Create a folder and store all the programs in the folder.

Notes

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 23

Write a C++ program to accept an integer array of size 10. Display the squares of all the elements by using a pointer to the array.

Objective

After completion of this practical you will be able to use pointers and arrays.

Software required

- Turbo C++

Pre-requisite

- Basic knowledge of pointers is required.

Procedure

- Accept an integer array of size 10.
- Create a pointer using * operator.
- Use a pointer to the array.
- Display the square of all elements while traversing the arrays using a pointer.

Dos and Don'ts

- Add your name and enrolment number as a comment on the top of the program
- Save the program with proper filename having .cpp extension
- Create a folder and store all the programs in the folder.

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 24

Define a class LIBRARY with the following members:

Members	Data type
book_id	integer
author_name	string
issue_date	string
fine	integer
return_date	string

Also define appropriate member functions to accept and display the data members of the class.

Write a C++ program to accept and write data of the class LIBRARY to the file "LIB.DAT". The program should also be able to display data stored in the file.

Objective

After completion of this practical you will be able to read and write from / to data files.

Software required

- Turbo C++

Pre-requisite

- Basic knowledge of opening data files using fstream objects is required. Learner should also know about read ( ), write ( ) and close ( ).

Procedure

- Open the file using constructor open ( ).
- Accept and display the data by calling the public member of the class.

- Use read ( ) and write ( ) to read from and write to the data file respectively.
- Close the file using close ( ).

Dos and Don'ts

- Add your name and enrolment number as a comment on the top of the program.
- Save the program with proper filename having .cpp extension.
- Create a folder and store all the programs in the folder.

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 25

Gita is running a departmental store in her colony. She wants to computerize the billing system. Consider a class containing suitable members and associated functions. The program should store details of all the bills in the file “BILL.dat”. Also a discount of 10% on the net bill amount should be given to all the customers whose net bill amount exceeds Rs 2000.

Objective

After completion of this practical you will be able to use data files.

Software required

- Turbo C++

Pre-requisite

- Basic knowledge of opening data files using fstream objects is required. Learner should also know about read ( ), write ( ) and close ( ).

Procedure

- Open the file using constructor open ().
- Accept and display the data by calling the public member of the class.
- Use read and write function.
- Close the file.

Dos and Don’ts

- Add your name and enrolment number as a comment on the top of the program.
- Save the program with proper filename having .cpp extension.
- Create a folder and store all the programs in the folder.

Notes

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 26

Create a learner table using SQL commands in the following format

Table Name is Learner

Column Name	Data Type	Size
Roll	Varchar2	10
Name	char	30
Address	Varchar2	50

Objective

After completion of this practical you will be able to create a new table using SQL.

Software required

- MySQL

Pre-requisite

- Learners should have knowledge of SQL commands.

Procedure

- Use following syntax for creating table

```
CREATE TABLE table_name (
        column1 datatype,
        column2  datatype,
        column3 datatype,
        ....
);
```

Dos and Don’ts

- Create and use database before creating the table.

Learner’s Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor’s signature

Practical 27

Add new column in existing table using SQL commands.

Table Name is Learner

Existing Table

Column Name	Data Type	Size
Roll	Varchar2	10
Name	char	30
Address	Varchar2	50

After adding new column, table should have following format

Column Name	**Data Type**	**Size**
Roll	Varchar2	10
Name	char	30
Address	Varchar2	50
DOB	date	

Objective

After completion of this practical you will be able to alter the table using SQL.

Software required

- MySQL

Pre-requisite

- Learners should have knowledge of SQL commands.

Procedure

- Use following syntax to add new column in table

```
ALTER TABLE <table_name>
      ADD (<NewColumnName> <Data_Type>(<size>),......n),
```

Dos and Don'ts

- Create database and use it.

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 28

Modify column value in existing table using SQL commands.

Table Name is Learner

Existing Table

Column Name	Data Type	Size
Roll	Varchar2	10
Name	char	30
Address	Varchar2	50

After updating column data, table should have following data

Column Name	Data Type	Size
Roll	Varchar2	10
Name	Varchar2	40
Address	Varchar2	50

Objective

After completion of this practical you will be able to update the table data using SQL.

Software required

- MySQL

Pre-requisite

- Learners should have knowledge of SQL commands.

Procedure

- Use following syntax to update existing table data.

Notes

```
UPDATE TABLE <table_name>
SET (<column_name>) = <new value>
Where (<column_name>) = <existing_value>;
```

Dos and Don'ts

- Create database and use it.

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 29

Drop a column from existing table using SQL commands.

Table Name is Learner

Existing Table

Column Name	Data Type	Size
Roll	Varchar2	10
Name	char	30
Address	Varchar2	50
DOB	date	

After dropping column, table should have following format

Column Name	Data Type	Size
Roll	Varchar2	10
Name	char	30
Address	Varchar2	50

Objective

After completion of this practical you will be able to drop a column from table using SQL.

Software required

- MySQL

Pre-requisite

- Learners should have knowledge of SQL commands.

Procedure

- Use following syntax to drop column from table

  ALTER TABLE <table_name> DROP COLUMN <column_name>

  <NewDataType>(<NewSize>))

Dos and Don'ts

- Create database first then use it.

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 30

Table Name is Learner

Column Name	Data Type	Size
Roll	Varchar2	10
Name	char	30
Address	Varchar2	50
DOB	date	

Do following activity on above table

1. Select roll and name from above the table.
2. Eliminate duplicate rows from above the table.

Objective

After completion of this practical you will be able to remove duplicate rows from database.

Software required

- MySQL

Pre-requisite

- Learners should have knowledge of SQL commands.

Procedure

- Use SELECT command for selecting roll and name from Table.
- Use DISTINCT command for eliminating rows.

Dos and Don’ts

- Create database and use it.

Learner’s Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor’s signature

Practical 31

Sort the following table's records in (alphabetical order from A to Z) ascending order by Name column

Table Name is Learner

Column Name	Data Type	Size
Roll	Varchar2	10
Name	char	30
Address	Varchar2	50
DOB	date	

Objective

After completion of this practical you will be able to arrange database in ascending/descending order.

Software required

- MySQL

Pre-requisite

- Learners should have knowledge of SQL commands.

Procedure

- Use ORDER BY command along with SELECT command.
- For descending order, use following syntax

```
ORDER BY Column name DESC
```

Dos and Don'ts

- Use ORDER BY column name for ascending order.

Notes

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 32

Perform DML commands UPDATE and DELETE on the following table.

Table Name is Learner

Roll	Name	Marks
111	aaaa	77
222	accc	66
333	beee	99

1. Update existing marks 77 to 88.
2. Delete third row from the table.
3. Delete all rows from the table.

Objective

After completion of this practical you will be able to update and delete the existing table contents.

Software required

- MySQL

Pre-requisite

- Learners should have knowledge of SQL commands.

Procedure

- Use UPDATE command to modify data values in table.
- Use DELETE command to modify data values in table.

Dos and Don'ts

- Delete only the content of a table.

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 33

Perform DML commands INSERT on the following table.

Table Name is Learner

Roll	Name	Marks
111	aaaa	77
222	accc	66
333	beee	99

1. Insert new row (Roll=555, Name=vvvv, Marks=89)

Objective

After completion of this practical you will be able to insert new values in the existing table.

Software required

- MySQL

Pre-requisite

- Learners should have knowledge of SQL commands.

Procedure

- Use INSERT INTO command for adding new row.

Dos and Don'ts

- Insert in an existing table.

Notes

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 34

Create an HTML document which will display the ordered list of practical mentioned in this manual.

Objective

After completion of this practical you will be able to create ordered list of items.

Software required

- Notepad.
- Internet Explorer / Google Chrome/ Mozilla Firefox or any other browser software.

Pre-requisite

- Theoretical knowledge of HTML tags.

Procedure

- Open Notepad.
- Use <OL> tag, type the list of practical.
- Save it as home.html.

Dos and Don'ts

- Remember to save your file and the location where you are saving the file.

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 35

Create a web page that displays title “HELLO WORLD” and content “Delhi is the capital of India” in Italic and bold.

Objective

After completion of this practical you will be able to use title, italic and bold tags.

Software required

- Notepad.
- Internet Explorer / Google Chrome/ Mozilla Firefox or any other browser software.

Pre-requisite

- Theoretical knowledge of HTML tags.

Procedure

- Open Notepad.
- Use following tags

  a. <title>……..</title>

  b. <i>…..</i>

  c. <b>….</b>

Dos and Don’ts

- Remember to save your file and the location where you are saving the file.

Learner’s Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor’s signature

Practical 36

Create a web page that contains the following table:

Roll	Name	Marks
111	aaaa	77
222	accc	66
333	beee	99

Objective

After completion of this practical you will be able to create table in webpage.

Software required

- Notepad
- Internet Explorer / Google Chrome/ Mozilla Firefox or any other browser software

Pre-requisite

- Theoretical knowledge of HTML tags.

Procedure

- Open Notepad.
- Use <tr> tag for rows.
- Use <td> tag for columns.

Dos and Don’ts

- Remember to save your file and the location where you are saving the file.

Learner’s Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor’s signature

Practical 37

Add an image on existing web page.

Objective

After completion of this practical you will be able to add image in your webpage.

Software required

- Notepad.
- Internet Explorer / Google Chrome/ Mozilla Firefox or any other browser software.

Pre-requisite

- Theoretical knowledge of HTML tags.

Procedure

- Open Notepad.
- Use following tags

```
<img src= “dinosaur.jpg”>
```

- If image is stored at different place, give complete path of image location after src.

Dos and Don’ts

- Remember to save your image file and the location where you are saving the file.

Learner’s Observations

....

....

....

Notes

....

....

....

....

....

....

....

....

....

Instructor's signature

Practical 38

Create a C++ project on any topic of your interest. The project should have a menu driven opening screen to accept, display and search records. For example, if you are doing a project of Library Management System, the opening menu may be as follows:

1. Enter Data
2. Display All Records
3. Search a Record
4. Exit

By clicking on any menu item, the corresponding task should be performed.

After creating and running the project successfully, each learner has to make a project file using OpenOffice Writer containing the following items:

- Title page containing name of the project, learner name and enrolment number
- Acknowledgement
- Introduction to the topic
- Hardware/Software requirements
- Built-in functions (with header files) used in the project
- Coding
- Output Screens
- Bibliography

The file should be spiral bound and submitted during your practical examinations.

The learner should also create a presentation highlighting key features of his/her project and present it in the class.

Learner's Observations

....

....

....

....

....

....

....

....

....

....

....

....

Instructor's signature