

STES's  
**SINHGAD COLLEGE OF  
ENGINEERING**  
Vadgaon(Bk), Pune

**Department of Computer Engineering**



**LABORATORY MANUAL**

2022-23

**DATABASE MANAGEMENT SYSTEM  
TE-COMPUTER ENGINEERING**

**SEMESTER-I**

Subject Code: **310241**

Roll No :

Batch :

Seat No:

**TEACHING SCHEME**

Lectures: 3Hrs/Week

Practical: 2 Hrs./Week

**EXAMINATION SCHEME**

Practical: 25 Marks

Term Work: 25 Marks

**-: Name of Faculty:-**

Prof. M.D. Sale

Prof. H.E.Chaudhari

Prof. A. V. Dirgule



Sinhgad Technical Educational Society"s

**SINHGAD COLLEGE OF ENGINEERING  
PUNE**

**CERTIFICATE**

*This is to certify that*

*Mr. / Miss \_\_\_\_\_,*

*Of Class \_\_\_\_\_ Roll No. \_\_\_\_\_ Has completed all the  
practical work in the subject \_\_\_\_\_  
satisfactorily in the Department of \_\_\_\_\_  
as prescribed by Savitribai Phule Pune University, in the  
academic year 2022 - 2023*

Staff In-charge

Head of the Department

Principal

**Assignment No.: 03****Problem Statement:**

Write a program to create a Dynamic Link Library for any mathematical operations (arithmetic, trigonometric and string operation) and write an application program to test it. (Java Native Interface/Use VB/VC++)

**Objectives:**

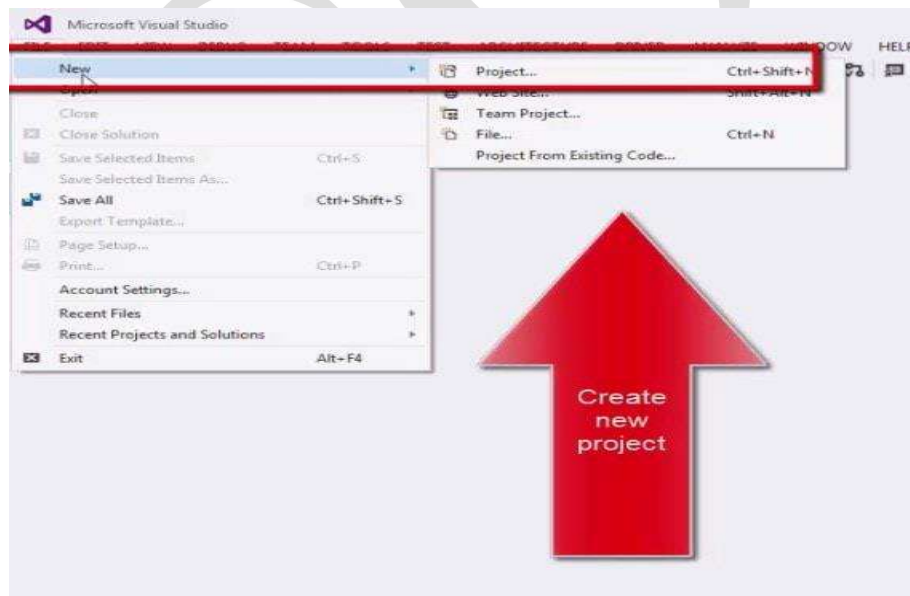
1. To study and understand concept of DLL.
2. To understand VC++.
3. To implement DLL using VC++.

**Theory:**

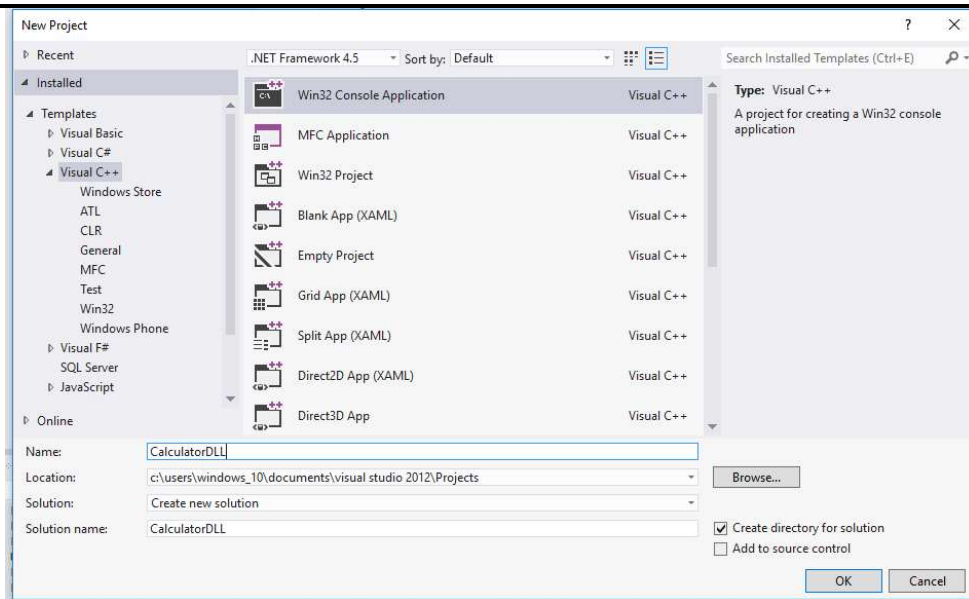
1. What is DLL? Significance of DLL. Advantages/ Disadvantages of DLL?
2. Explain Import and Export functions used in DLL?

**Algorithm/Flow Chart:****Steps to create DLL in C++**

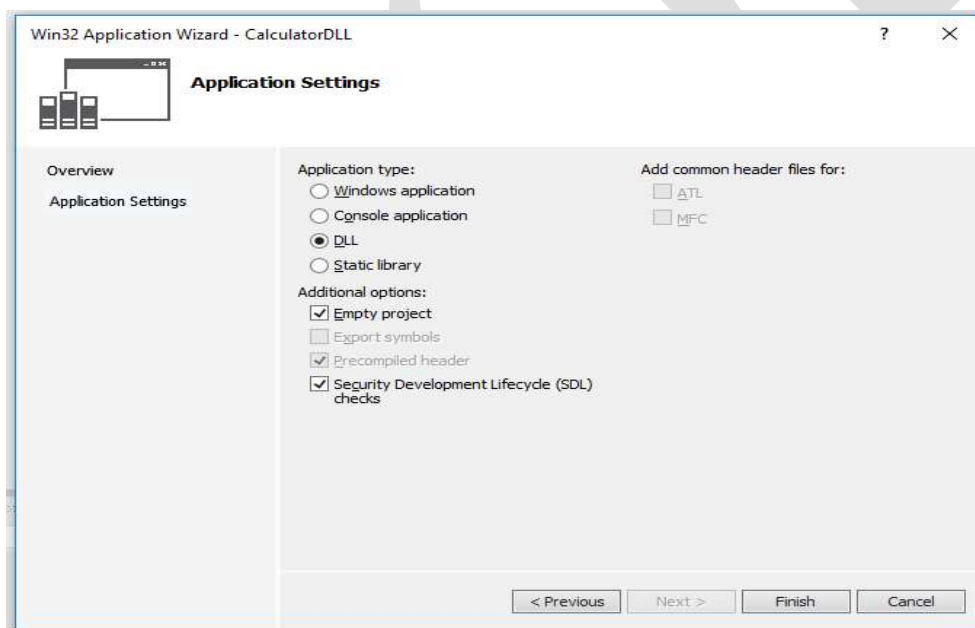
- Open the visual studio and click on the menu bar to create a new project. See the below Image.



- After selecting the new project, a new dialog box will be open, here select the project type Win32 and give the name to the DLL project.



- On the Overview page of the Win32 Application Wizard dialog box, choose the Next button. After clicking the next button a new window will open. It is Application setting window here we will select the type of the application and click on the finish button to create the DLL project.



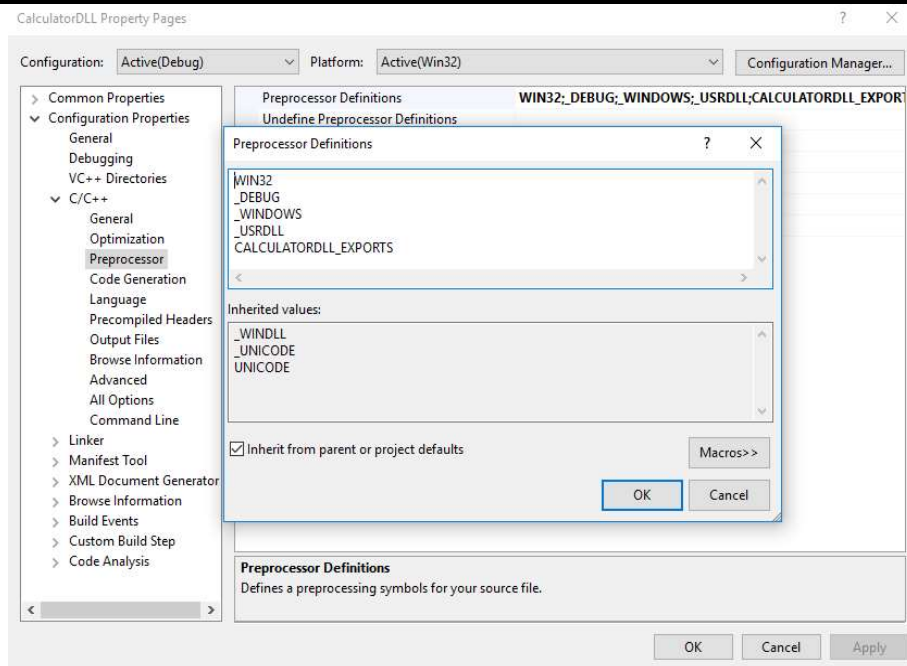
- After creating the DLL project you have to add the header files and source file as per your requirements. Here I am adding only one header file.



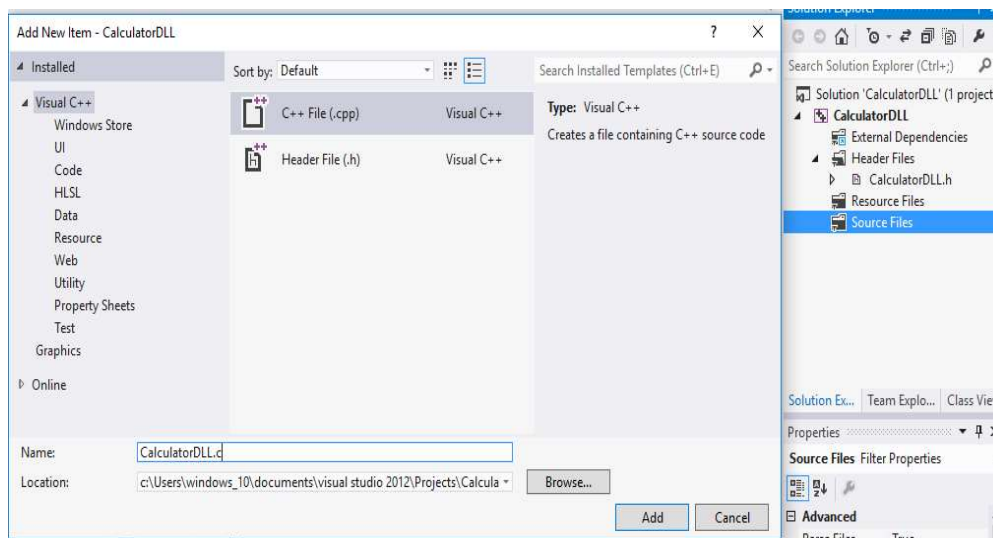
- When you have created the header file then write the desired content as per the requirements. Here I am creating a library that performs some basic arithmetic operation like addition.

```
#ifndef _CALCULATORDLL_h_
#define _CALCULATORDLL_h_
#ifdef CALCULATORDLL_EXPORTS
#define CALCULATORDLL_API __declspec(dllexport)
#else
#define CALCULATORDLL_API __declspec(dllimport)
#endif
CALCULATORDLL_API int Addition(int x,int y);
#endif
```

**Note:** When you have created a DLL project then automatically PROJECTNAME\_EXPORTS is defined in preprocessor symbols of the DLL project. In this example, CALCULATIONDLL\_EXPORTS is defined when your CALCULATIONDLL DLL project is built.



- Now it's time to define your class member function in the source file. Here I am defining all member functions in CalculatorDLL.C file.



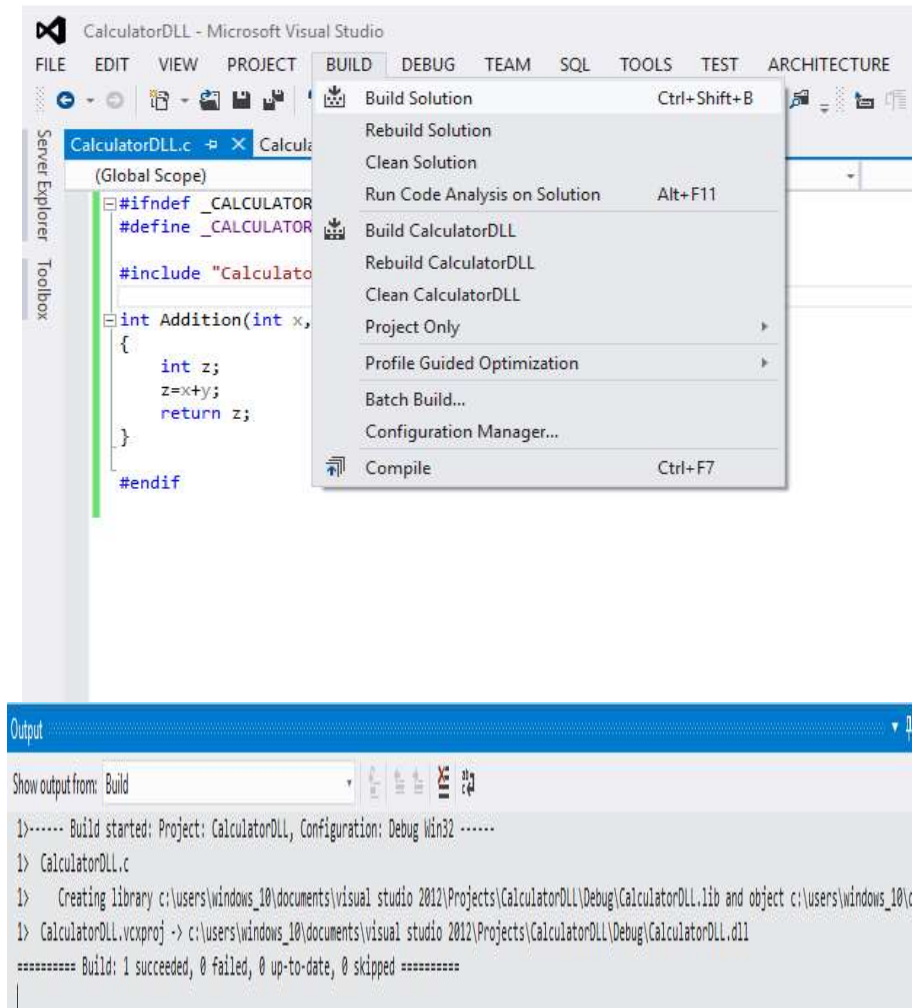
```
#ifndef _CALCULATORDLL_c_
#define _CALCULATORDLL_c_
```

```
#include "CalculatorDLL.h"
```

```
int Addition(int x,int y)
{
    int z;
    z=x+y;
    return z;
}
```

#endif

Now source and header files are added to the DLL project, to create the DLL and lib just build the DLL project. If everything is fine and your DLL project compiles perfectly without any error then a DLL and .lib file will be generated.

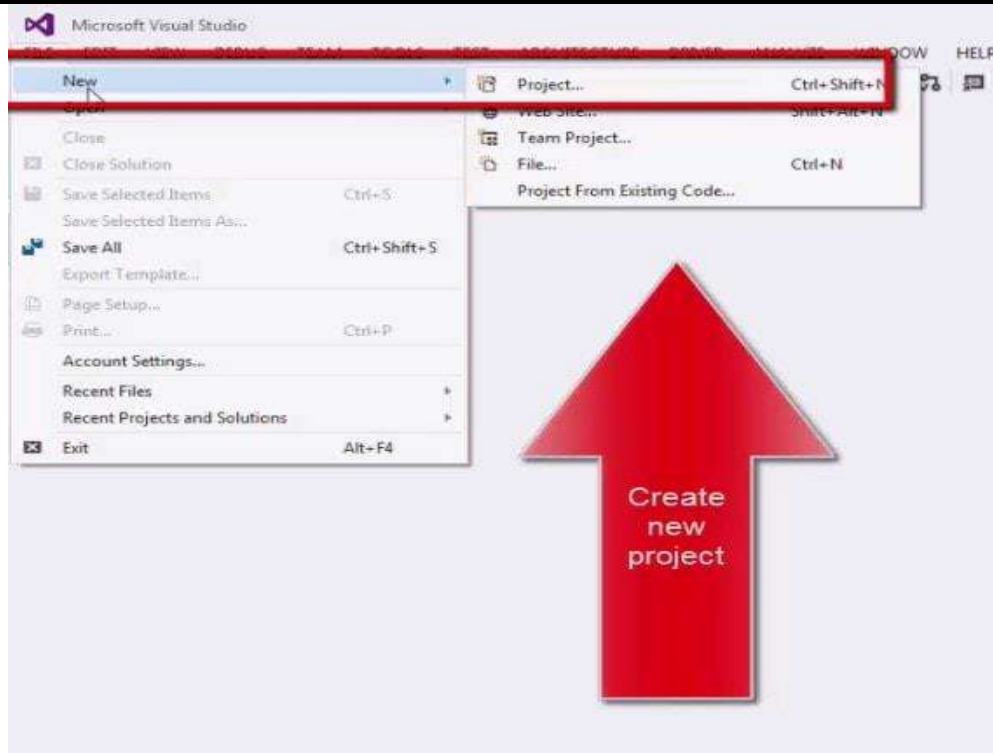


### Steps to create a C ++ Application

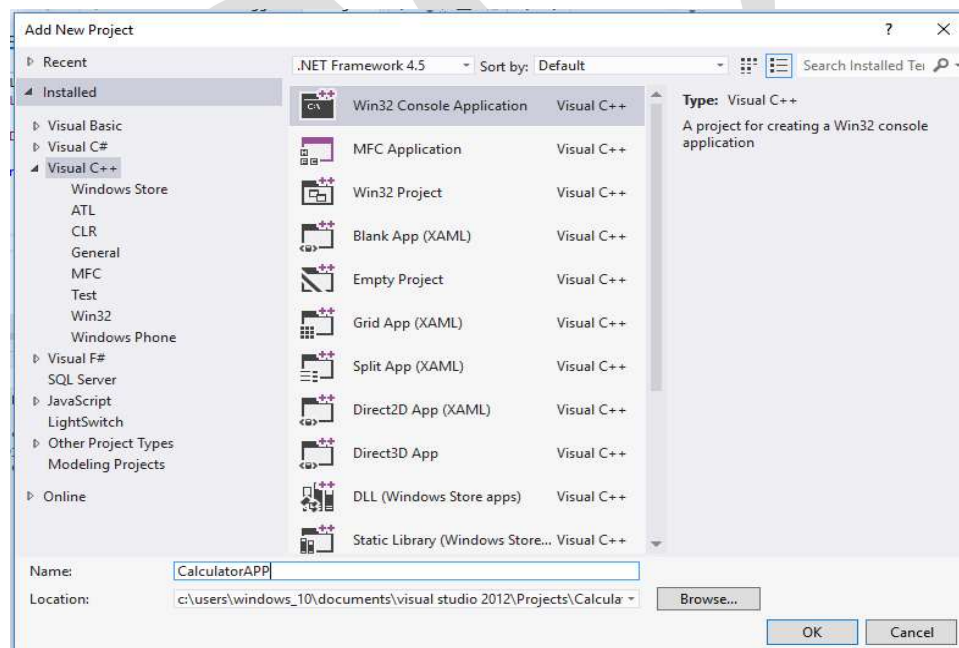
Here I am creating a c++ application that will use the created DLL.

- Click on the menu bar to create a new c++ Application project that uses the DLL which I have created just now.



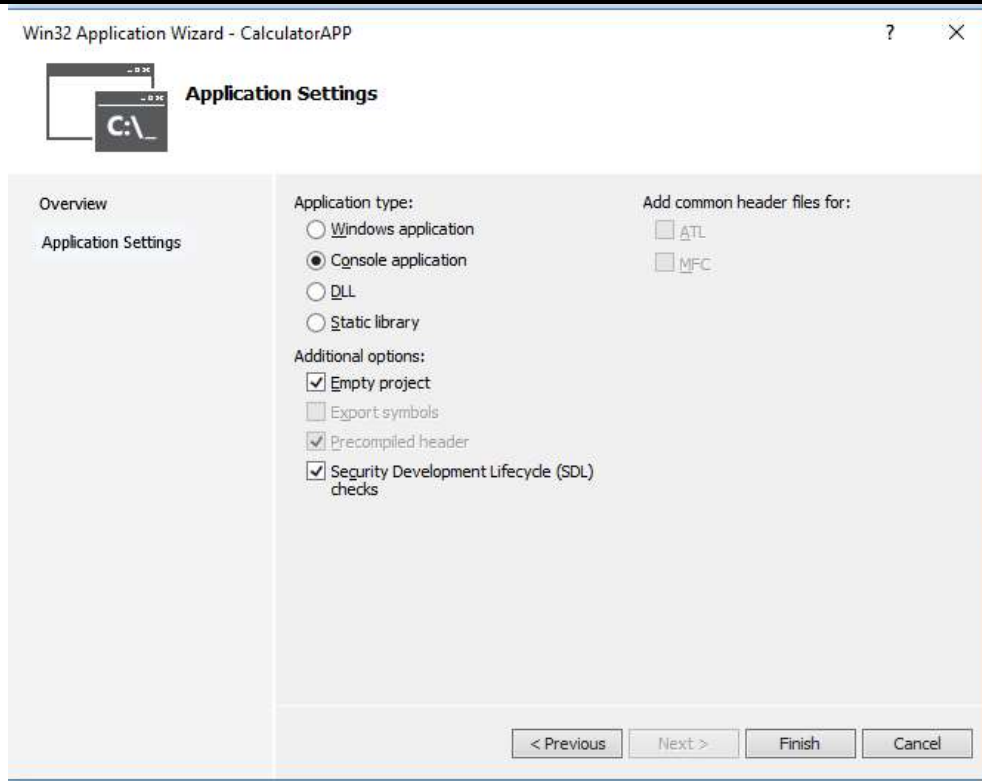


- After selecting the new project a new dialog box will be open, here select the project type Win32 Console Application and give the name to the App project.



- On the Overview page of the Win32 Application Wizard dialog box, choose the Next button. After clicking the next button a new window will open. It is the Application setting window here we will select the type of the application and click on the finish button to create the c++ Console Application project.



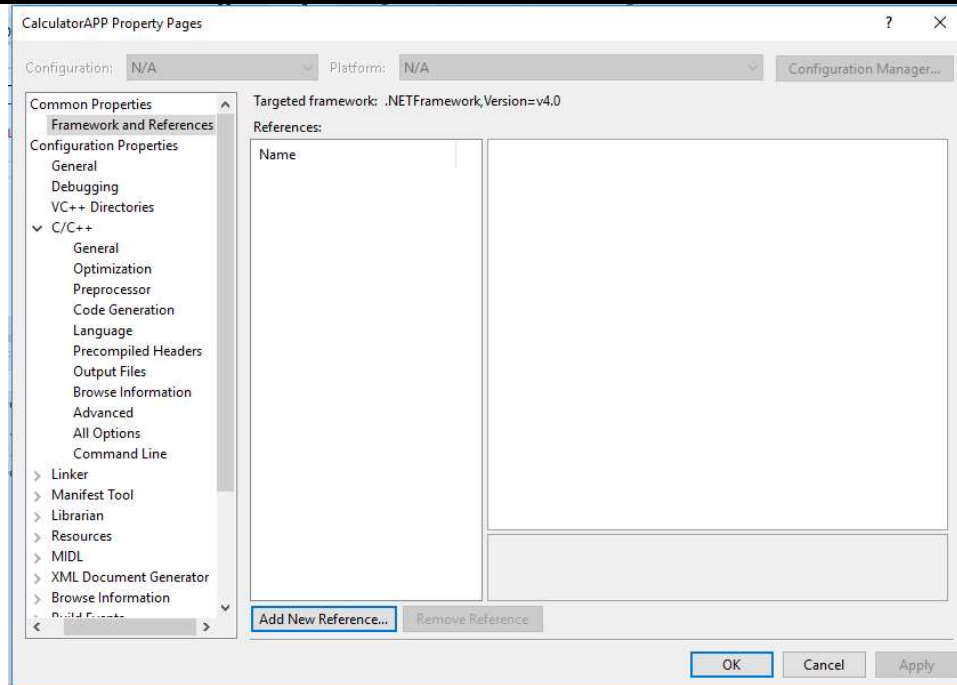


Now your C++ application project is ready to use the DLL ( Dynamic linking library).

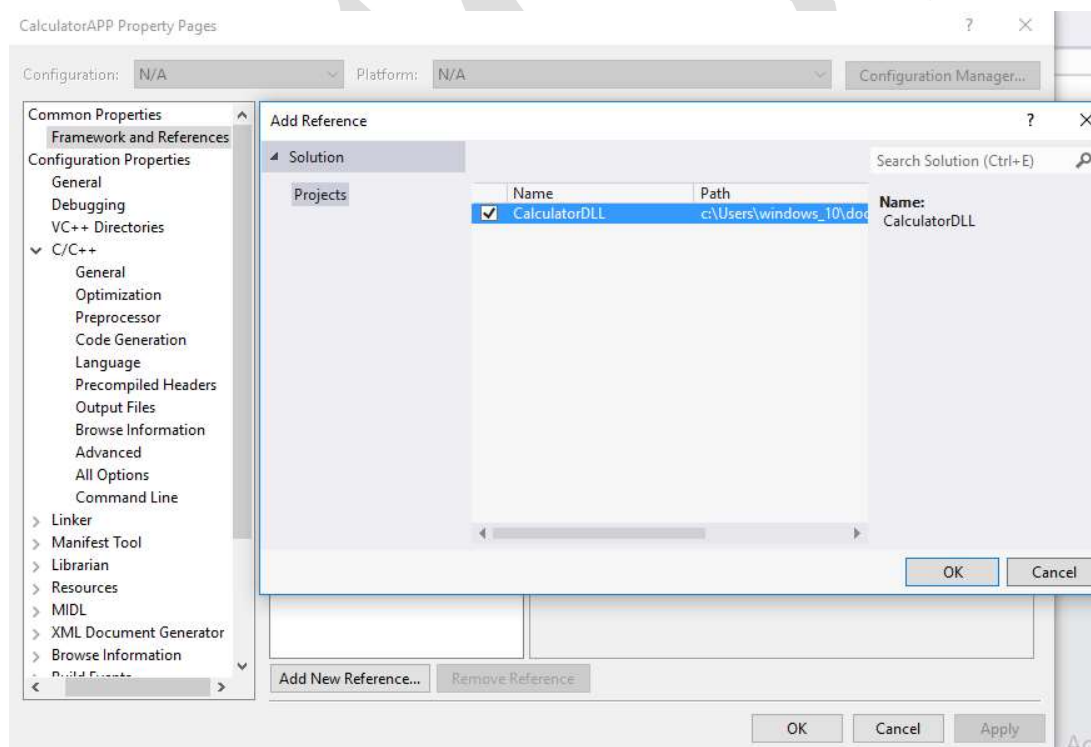
### How to Link DLL with c++ Application

Here I am discussing simple steps to link the DLL project with the C++ Application project.

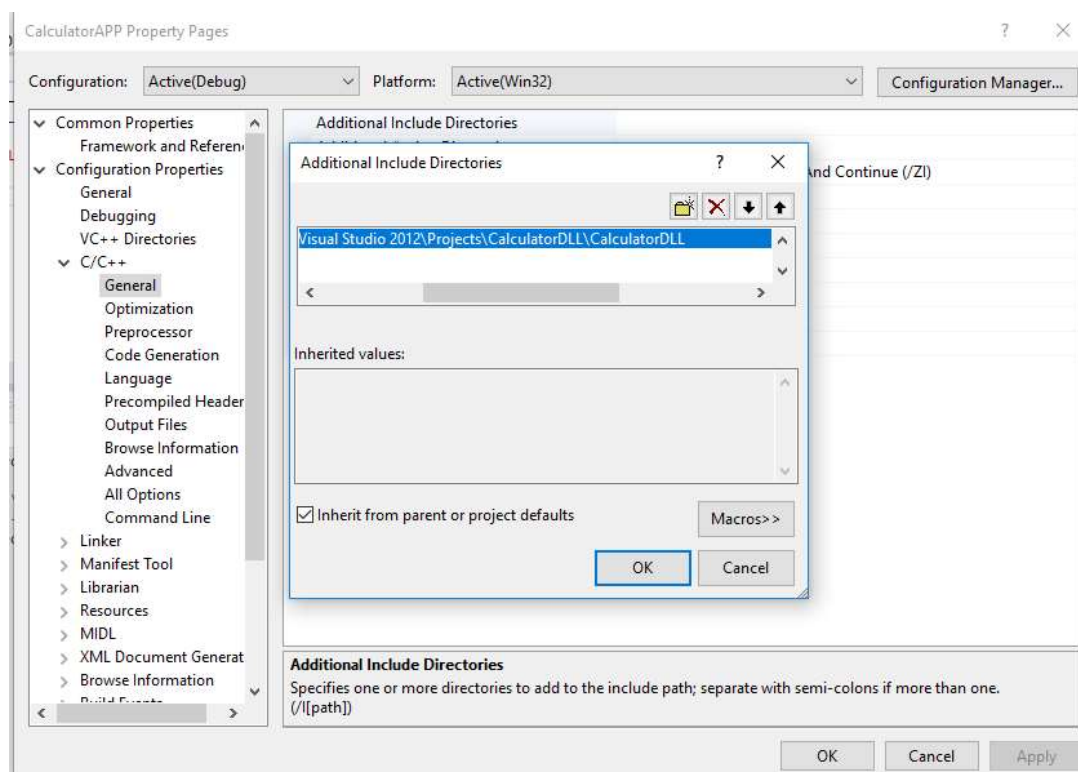
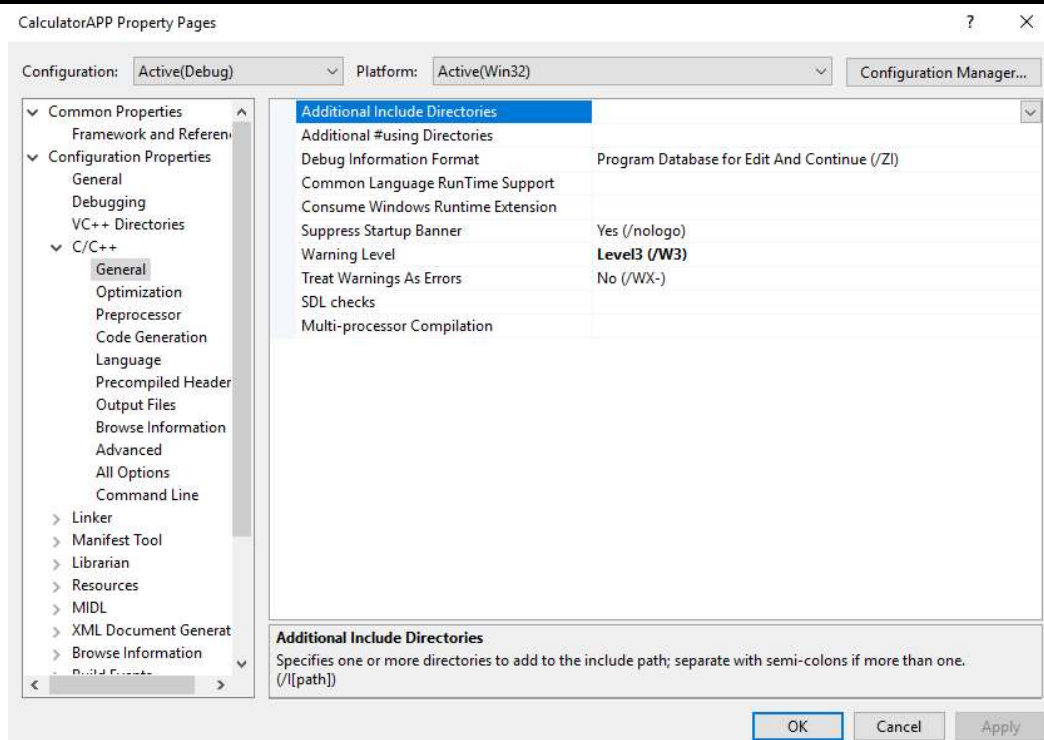
- When we have created the DLL and Application then after that we have to reference the DLL to the Application that makes the enable to Application to use the DLL function as per the requirement. To do this, under the CalculatorAPP project in Solution Explorer, select the References item. On the menu bar, choose Project, Add Reference.

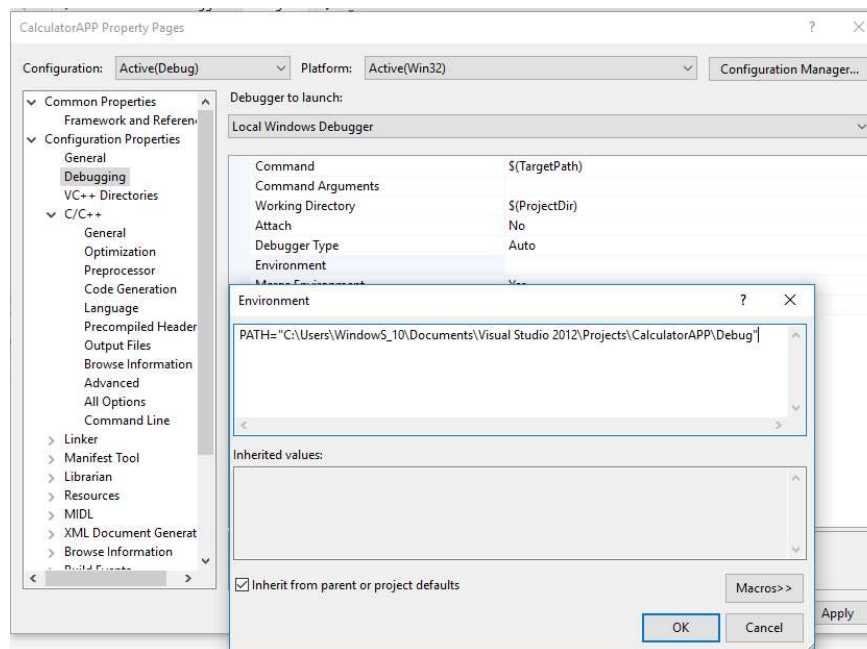
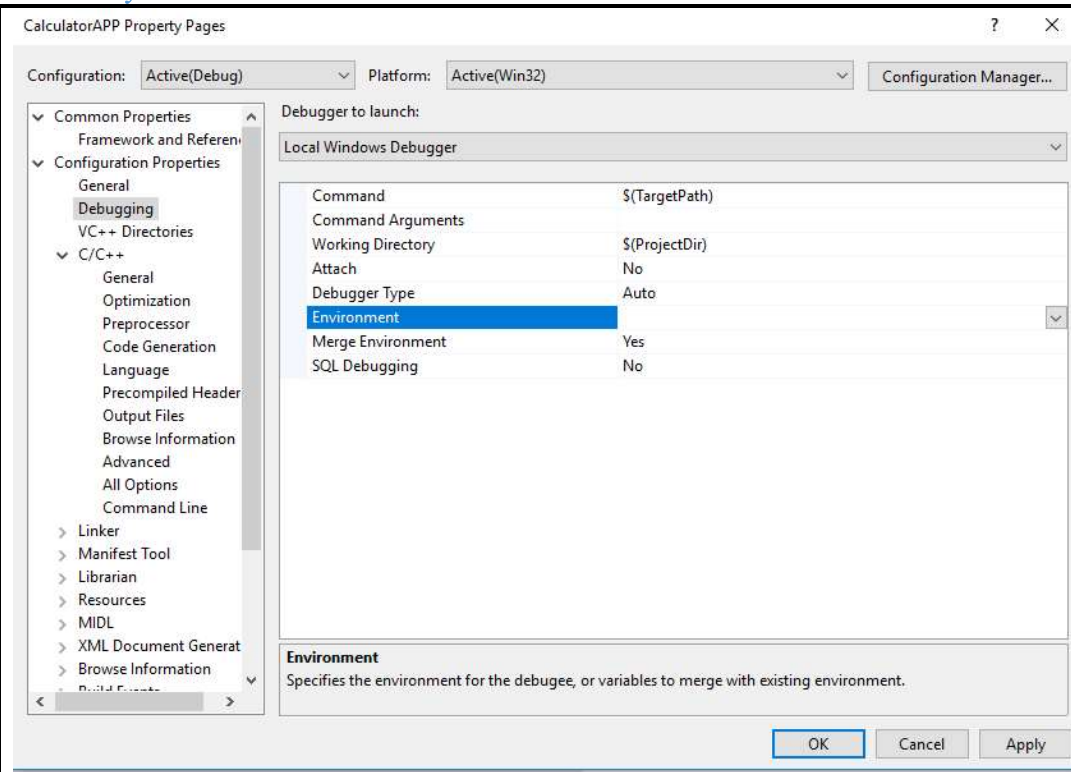


- When you click on the Add new Reference then a dialog box will be open which has the lists of the library that you can reference. You need to just click on the check button to the required library. Here only one library is showing in the dialog box.

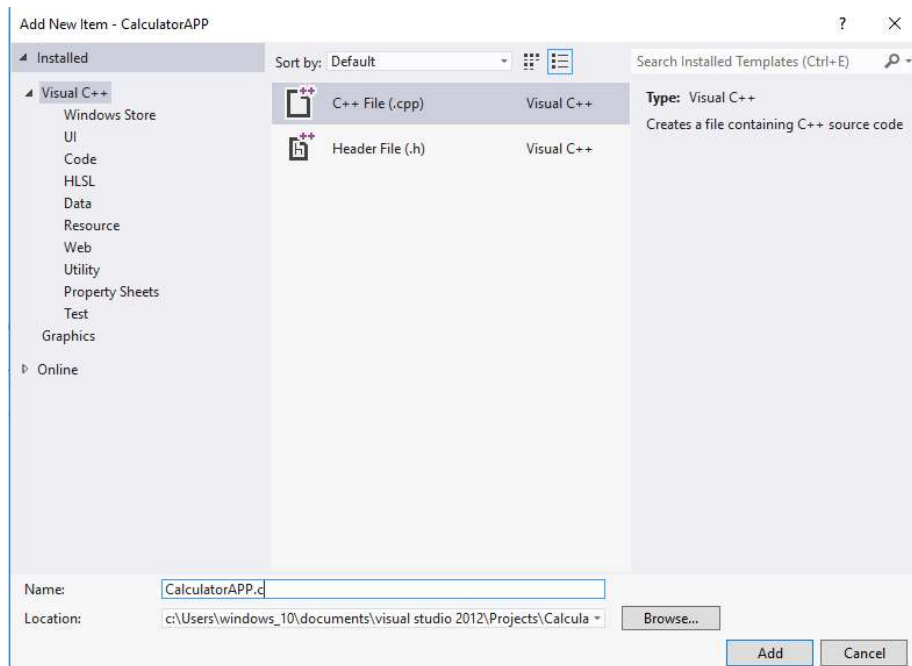
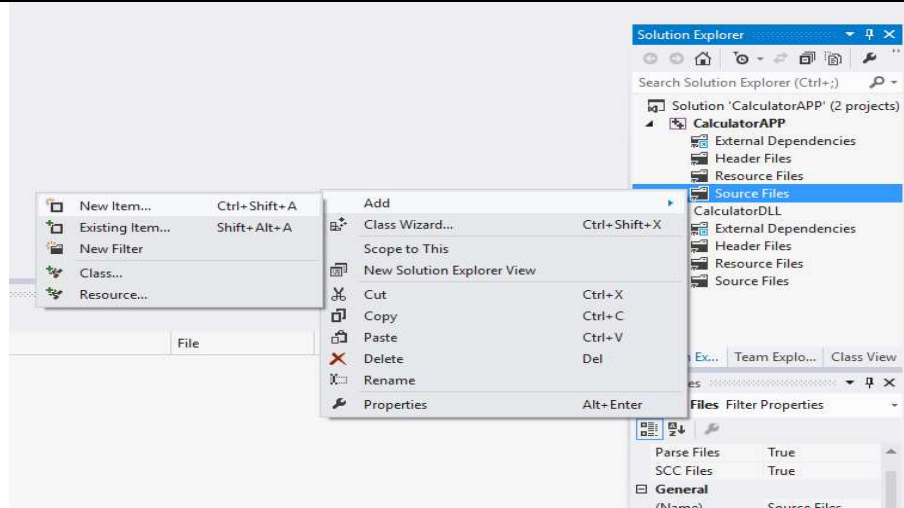


- Now your created library is linked with the created Application, but before using the DLL in Application you have to add the DLL header file. We just reference the DLL header file to give the path of original DLL header files in Application project included directories path.





- Now it's time to define your class member function in the source file. Here I am calling all member functions in CalculatorAPP.C file.

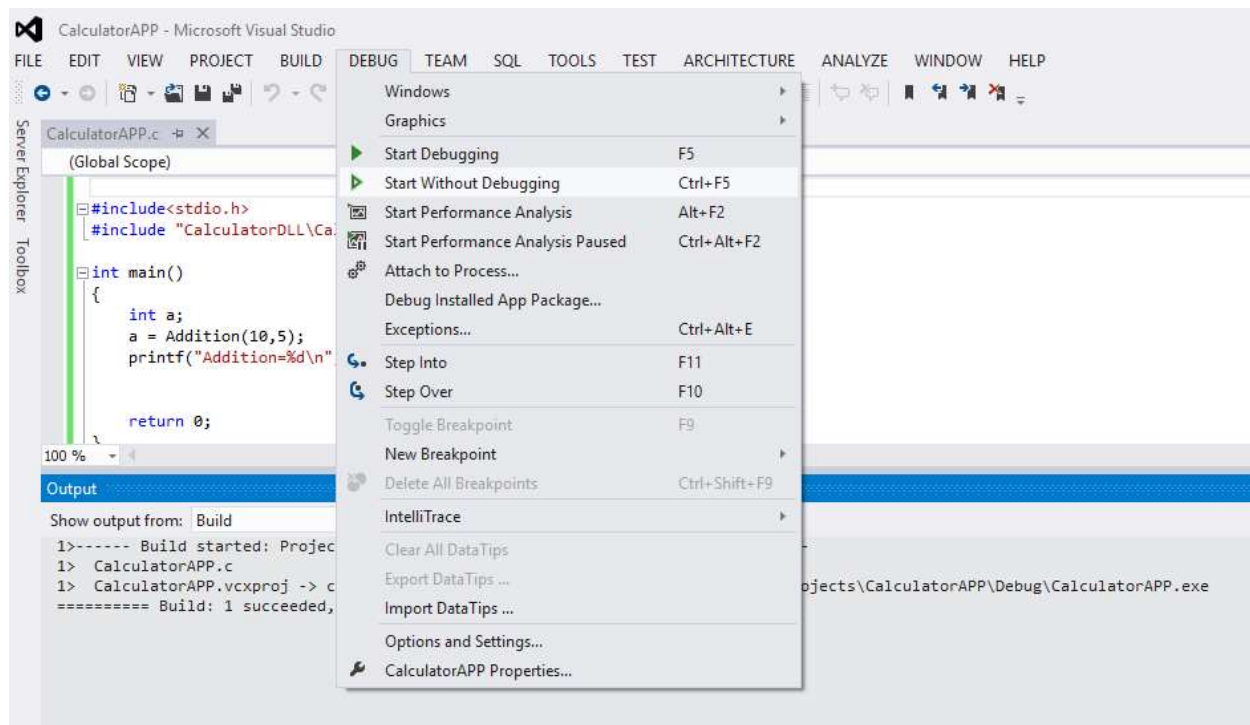
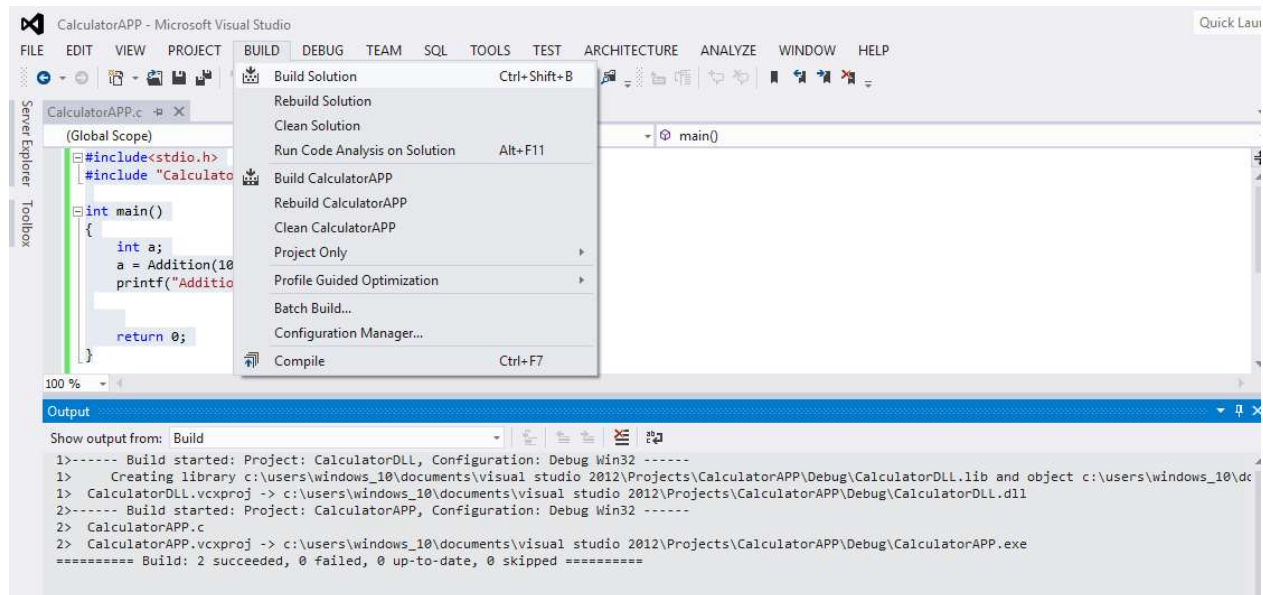


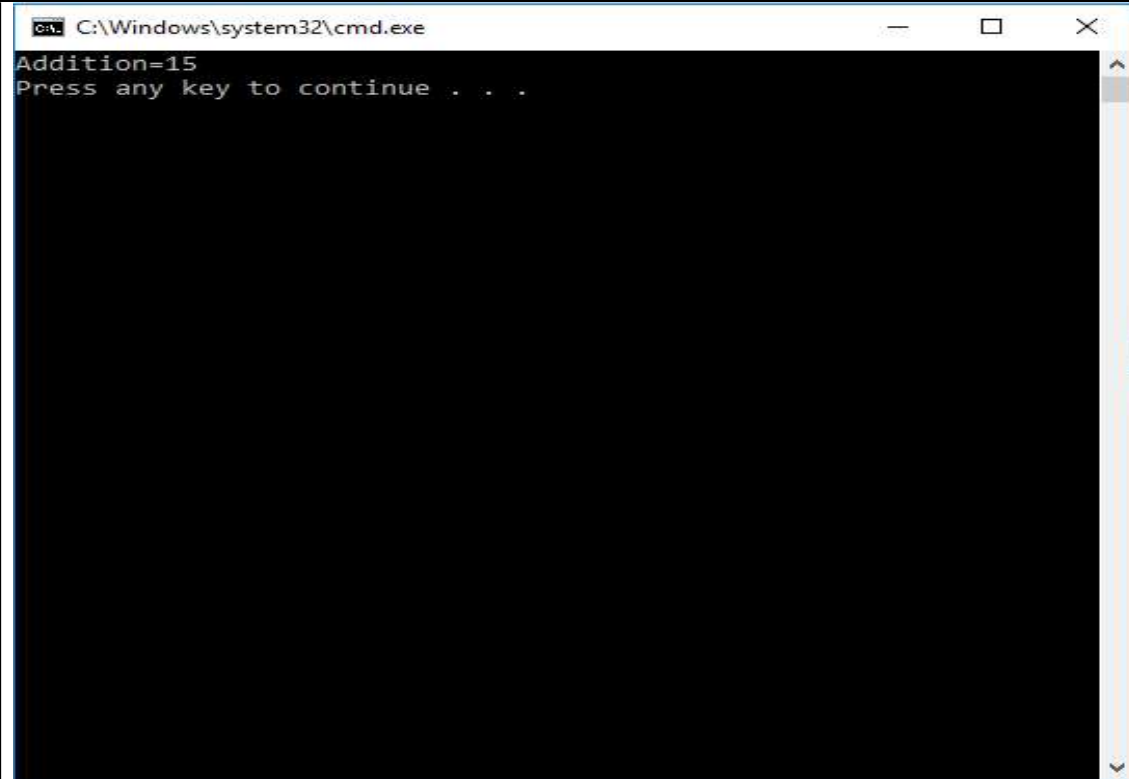
```
#include<stdio.h>
#include "CalculatorDLL\CalculatorDLL.h"
```

```
int main()
{
    int a;
    a = Addition(10,5);
    printf("Addition=%d\n",a);
}
```

```
return 0;
```

```
}
```





```
C:\Windows\system32\cmd.exe
Addition=15
Press any key to continue . . .
```

**Input:**

Enter Number1:10  
Enter Number2:05

**Output:**

Choose Operation:

- 1.Addition.
- 2.Substraction.
- 3.Multiplication.
- 4.Division.

Enter Your Choice:1

Addition:15

Choose Operation:

- 1.Addition.
- 2.Substraction.
- 3.Multiplication.
- 4.Division.

Enter Your Choice:2

Substraction:05

Choose Operation:

- 1.Addition.
- 2.Substraction.
- 3.Multiplication.
- 4.Division.

Enter Your Choice:3



**Assignment No.: 01****Problem Statements:**

Understanding the connectivity of Raspberry-Pi / Adriano with IR sensor. Write an application to detect obstacle and notify user using LEDs.

**Objectives:**

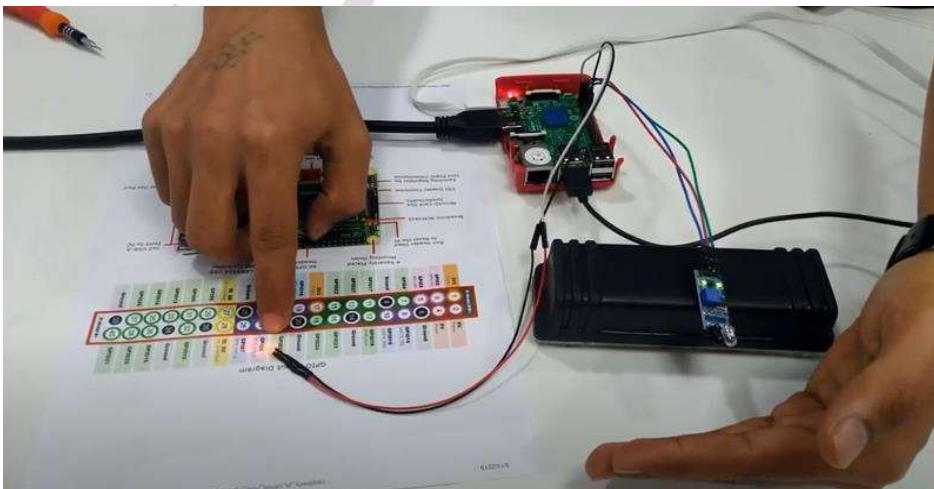
1. To understand the concept of Proximity sensor.
2. To interface Proximity sensor with Raspberry Pi model.
3. To program the Raspberry Pi model to detect the nearest object using proximity sensor and give indication through led.

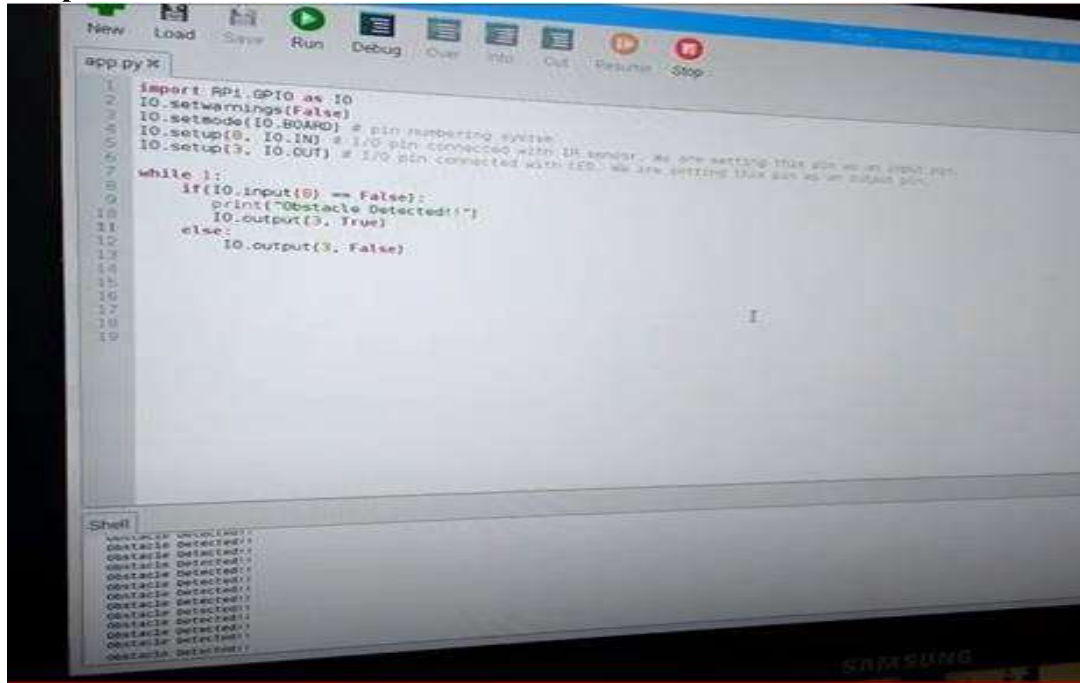
**Theory:**

1. What is Proximity sensor?
2. Explain Interface Diagram of Raspberry-Pi / Adriano with IR sensor
3. Explain assembling circuit steps?

**Algorithm/ Flow Chart:**

1. Import GPIO and Time library
2. Set mode i.e. GPIO.BOARD
3. Set GPIO pin '15' as Input
4. Set GPIO pin '16' as Input
5. Read input from GPIO pin '15'
6. Store the input value in the variable 'i'
7. If (i==1) then print the message as "Object is detected" and make the LED ON
8. If (i==0) then print the message as "No object detected" and make the LED OFF

**Input:**

**Output:**

```
app.py x
1 import RPi.GPIO as IO
2 IO.setwarnings(False)
3 IO.setmode(IO.BOARD) # pin numbering system
4 IO.setup(8, IO.IN) # I/O pin connected with IR sensor. We are setting this pin as an input pin.
5 IO.setup(3, IO.OUT) # I/O pin connected with LED. We are setting this pin as an output pin.
6
7 while 1:
8     if(IO.input(8) == False):
9         print("Obstacle Detected!")
10        IO.output(3, True)
11    else:
12        IO.output(3, False)
13
14
15
16
17
18
19
```

Shell

```
Obstacle Detected!
Obstacle Detected!
Obstacle Detected!
Obstacle Detected!
Obstacle Detected!
Obstacle Detected!
Obstacle Detected!
Obstacle Detected!
Obstacle Detected!
Obstacle Detected!
Obstacle Detected!
```

**Software Requirement:**

Raspbian OS (IDLE)

**Hardware Requirement:**

Raspberry Pi Board

Proximity sensor, Led, 330 ohm register

Monitor

**Conclusion:**

Successfully done the connectivity of Raspberry-Pi / Adriano with IR sensor. Tested connectivity by using LEDs to detect obstacle.

**Assignment No.: 02****Problem Statements:**

Understanding the connectivity of Raspberry-Pi /Beagle board circuit with temperature sensor. Write an application to read the environment temperature. If temperature crosses a threshold value, generate alerts using LEDs.

**Objectives:**

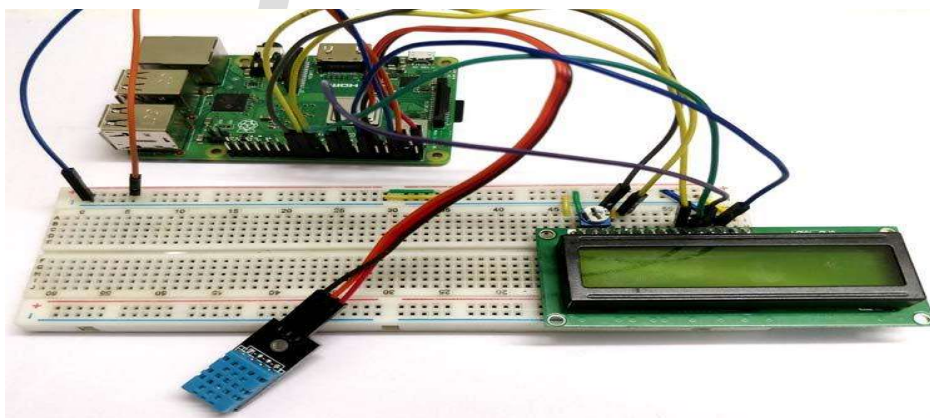
1. To understand the concept of Temperature-Humidity sensor (DHT11).
2. To interface Temperature-Humidity sensor with Raspberry Pi model.
3. To program the Raspberry Pi model to measure the real time Temperature and Humidity of the Environment.

**Theory:**

1. What is DHT11?
2. Explain Interface Diagram of Raspberry-Pi / Adriano with Temperature-Humidity Sensor?
3. Explain assembling circuit steps?

**Algorithm/ Flow Chart:**

1. Import GPIO, time and dht11 libraries.
2. Set all the warnings as False.
3. Set mode i.e. GPIO.BOARD
4. Read data using GPIO pin number 7 (dhtPin)
5. Write 'while loop' for displaying Temperature and Humidity values continuously
6. First Read the GPIO pin and Store the data in dhtValue Variable.
7. Print the temperature value.
8. Print the Humidity value.
9. Give delay of 1 second

**Input:**

### Problem Statements:

### Objectives:

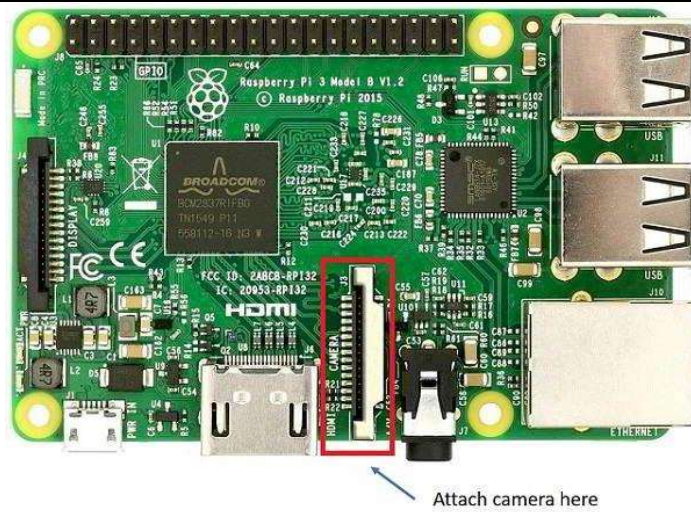
1. To understand the working of Raspberry Pi Camera .
2. To interface Raspberry Pi Camera with Raspberry Pi model.
3. To program the Raspberry Pi model to control the Raspberry Pi Camera Preview.
4. To program the Raspberry Pi model to capture still images from the Raspberry Pi Camera

**Algorithm/ Flow Chart:**

- a. To program the Raspberry Pi model to control the Raspberry Pi Camera Preview:
  1. Import Picamera library o Import time library o Create a variable(instance) of PiCamera class
  2. Display the camera preview on screen using the command “start\_preview()”.
  3. We can define 10 second delay to see the camera preview.
  4. To stop camera preview after 10 second, use the command “stop\_preview()”.
- b. To program the Raspberry Pi model to capture still images from the Raspberry Pi Camera 34.
  5. Import picamera library o Import time library.
  6. Create a variable(instance) of PiCamera class.
  7. Display the camera preview on screen using start\_preview().
  8. We can define 5 second delay to see the camera preview.
  9. Capture the image using camera.capture(‘path of the image. extension’).
  10. Then stop the camera preview using the command “stop\_preview()”.

[illegible]





### Output:

