Program 01

// ConsoleApplication7.cpp : Defines the entry point for the console application.

//

#include "stdafx.h"

#include<iostream>

using namespace std;

class Stack

{

int Array[5];

int top;

public:

Stack()

{

top = -1;

}

void Push(int GetValue)

{

if (top < 4)

Array[++top] = GetValue;

else

cout << "Stak is full\n";

}

int pop()

{

if (top > -1)

return Array[top--];

else

cout << "Stak is empty\n";

}

};

int main()

{

Stack Object;

Object.Push(10);

Object.Push(20);

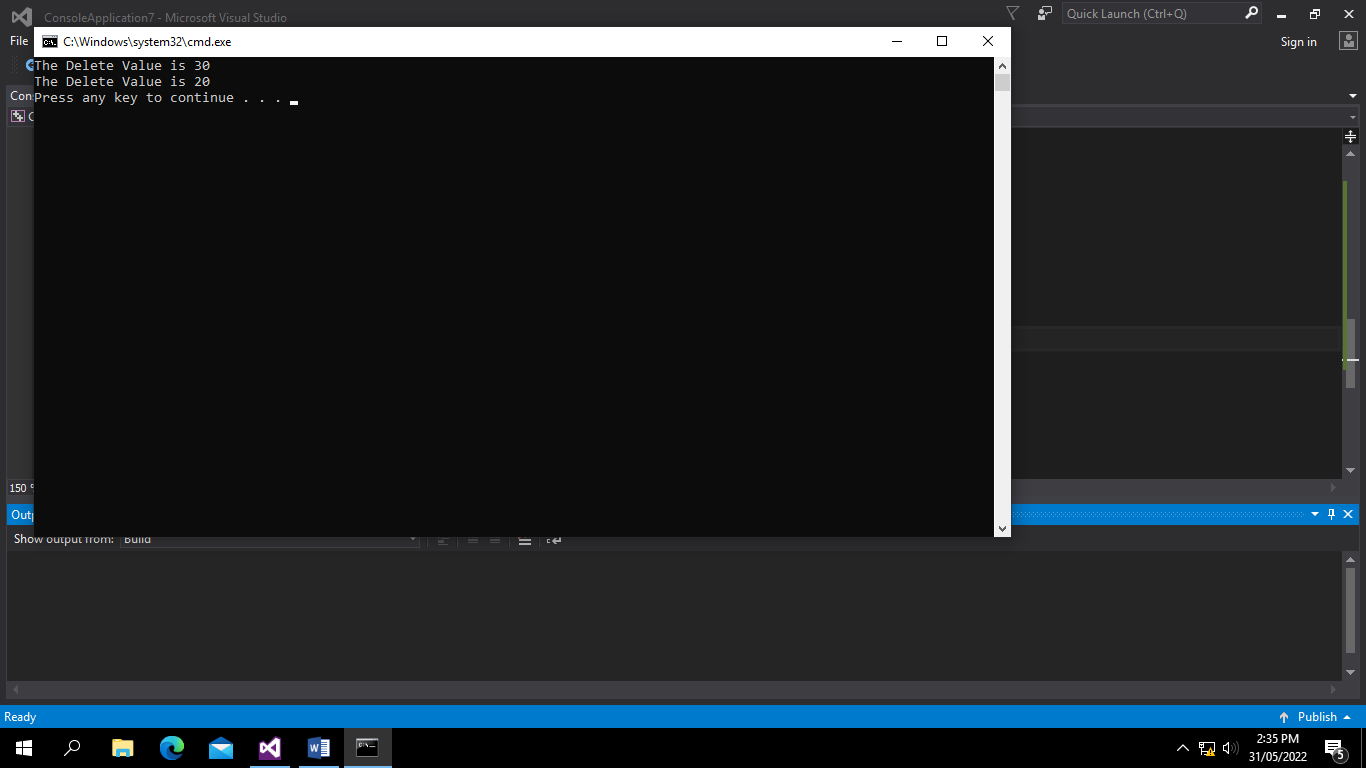
Object.Push(30);

cout << "The Delete Value is " << Object.pop() << endl;

cout << "The Delete Value is " << Object.pop() << endl;

return 0;

}



PROGRAM 02

// ConsoleApplication7.cpp : Defines the entry point for the console application.

//

#include "stdafx.h"

#include<iostream>

#include<string>

using namespace std;

class Stack

{

string Array[5];

int top;

public:

Stack()

{

top = -1;

}

void Push(string GetValue)

{

if (top < 4)

Array[++top] = GetValue;

else

cout << "Stak is full\n";

}

string pop()

{

if (top > -1)

return Array[top--];

else

cout << "Stak is empty\n";

}

};

int main()

{

Stack Object;

Object.Push("Ahmad");

Object.Push("Ali");

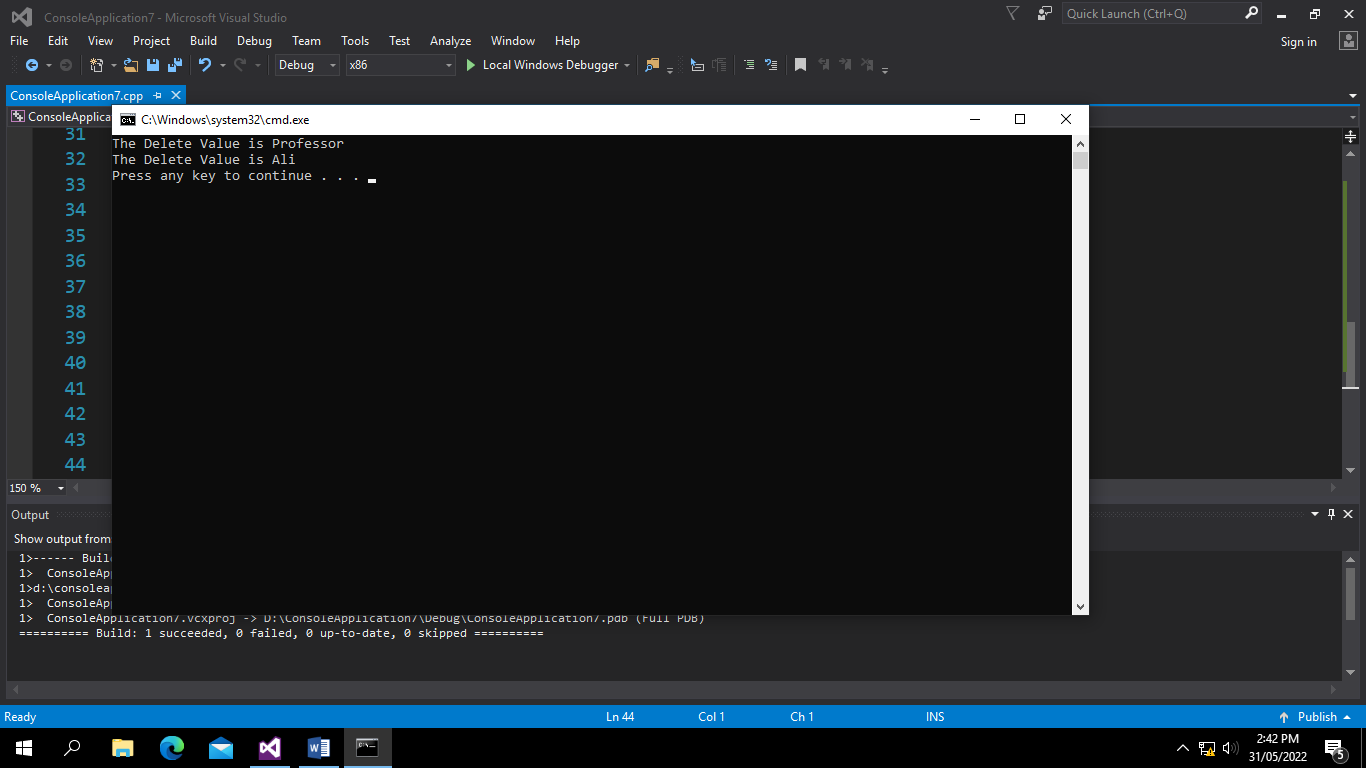
Object.Push("Professor");

cout << "The Delete Value is " << Object.pop() << endl;

cout << "The Delete Value is " << Object.pop() << endl;

return 0;

}



PROGRAM 03

// ConsoleApplication7.cpp : Defines the entry point for the console application.

//

#include "stdafx.h"

#include<iostream>

#include<string>

using namespace std;

class Stack

{

char Array[5];

int top;

public:

Stack()

{

top = -1;

}

void Push(char GetValue)

{

if (top < 4)

Array[++top] = GetValue;

else

cout << "Stak is full\n";

}

char pop()

{

if (top > -1)

return Array[top--];

else

cout << "Stak is empty\n";

}

};

int main()

{

Stack Object;

Object.Push('I');

Object.Push('P');

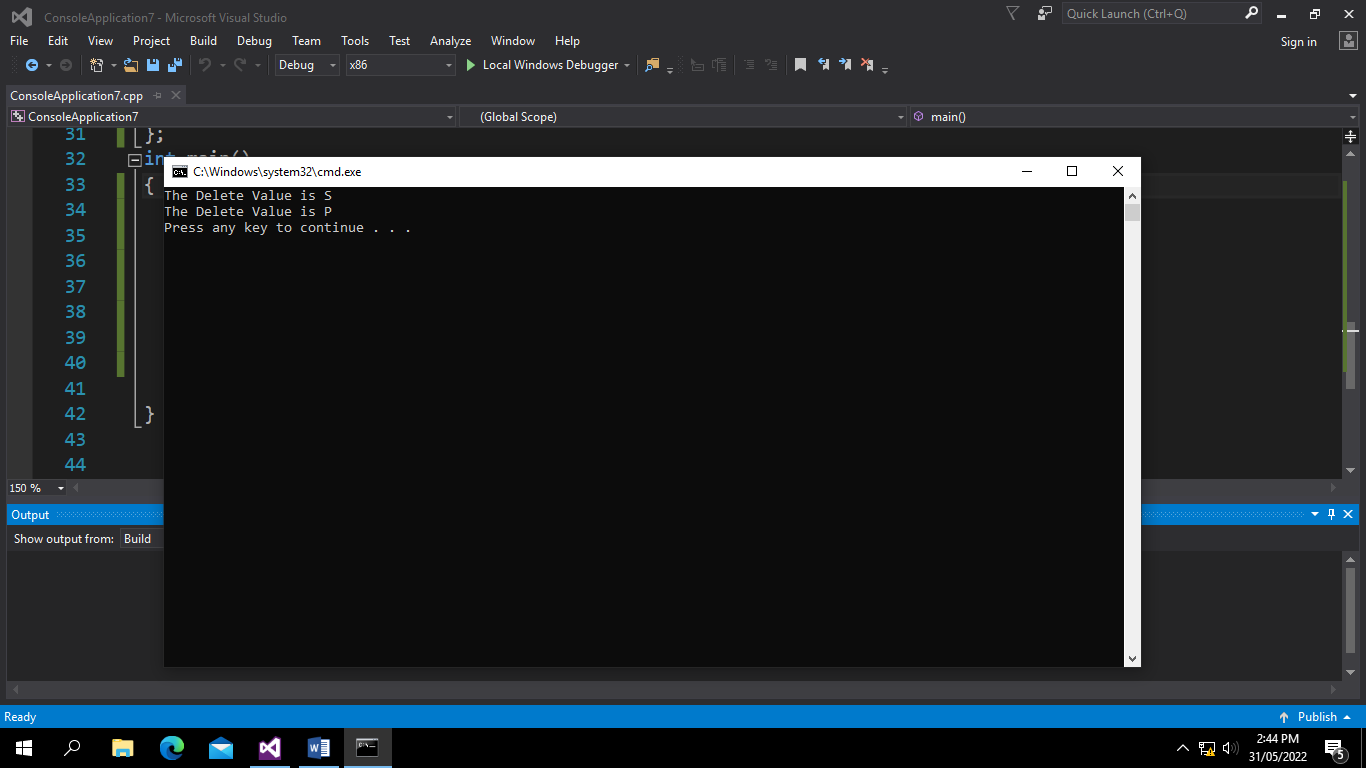
Object.Push('S');

cout << "The Delete Value is " << Object.pop() << endl;

cout << "The Delete Value is " << Object.pop() << endl;

return 0;

}



PROGRAM 04

// ConsoleApplication7.cpp : Defines the entry point for the console application.

//

#include "stdafx.h"

#include<iostream>

#include<string>

using namespace std;

class Student

{

int RollNo;

string Name;

float Cgpa;

public:

void input()

{

cout << "Etner the Roll No of Student : ";

cin >> RollNo;

cout << "Enter the Name of Student: ";

cin >> Name;

cout << "Enter the Cgpa of Student: ";

cin >> Cgpa;

}

void Display()

{

cout << "Roll NO of Student:" << RollNo << endl;

cout << "Name of Student: " << Name << endl;

cout << "Cgpa of Studnet: " << Cgpa << endl;

}

};

class Stack

{

Student Array[5];

int top;

public:

Stack()

{

top = -1;

}

void Push(Student GetValue)

{

if (top < 4)

Array[++top] = GetValue;

else

cout << "Stak is full\n";

}

Student pop()

{

if (top > -1)

return Array[top--];

else

cout << "Stak is empty\n";

}

};

int main()

{

Student StudentObject, StudentObject2, StudentObject3;

//StudentObject.input();

Stack Object;

StudentObject.input();

StudentObject2.input();

StudentObject3.input();

Object.Push(StudentObject);

Object.Push(StudentObject2);

Object.Push(StudentObject3);

cout << "\nNow Delete Last Value \n\n";

Student ObjectForPop, ObjectForPop2;

ObjectForPop = Object.pop();

ObjectForPop.Display();

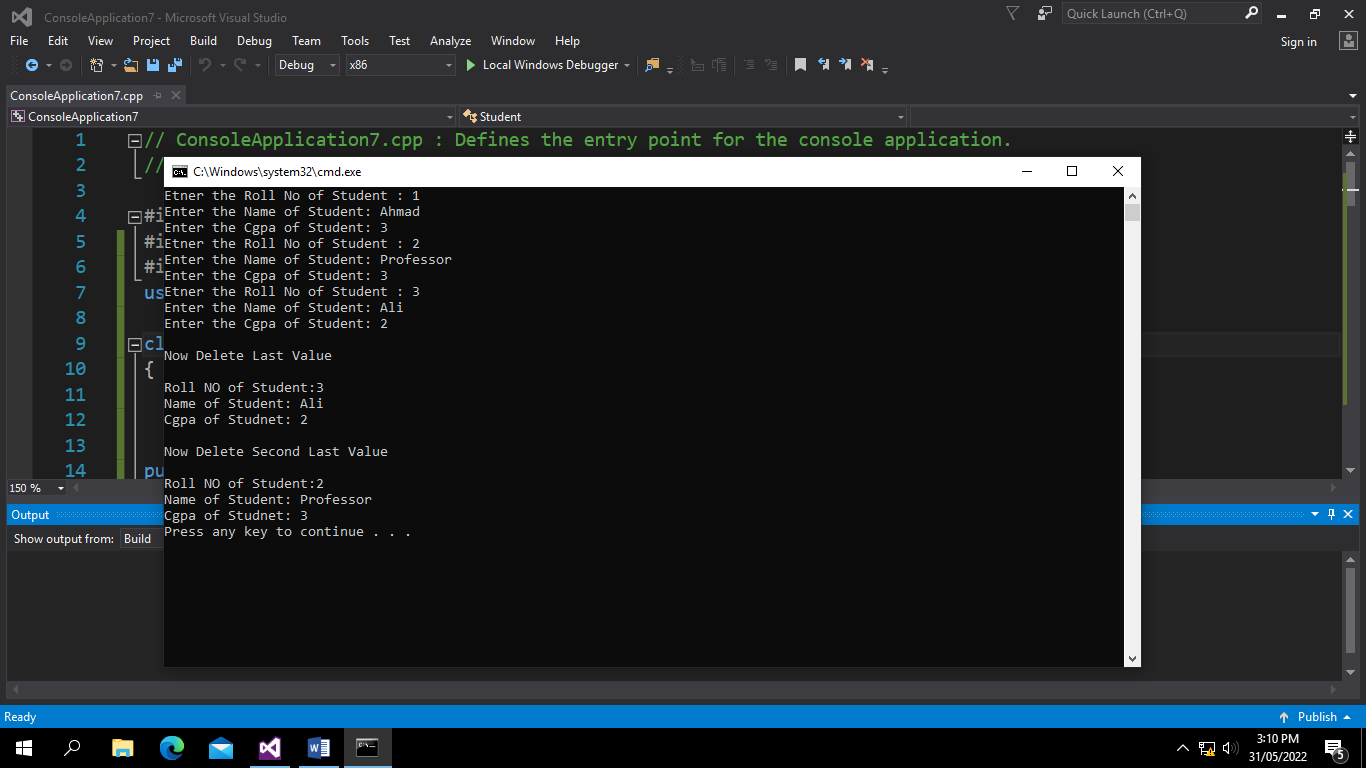
cout << "\nNow Delete Second Last Value \n\n";

ObjectForPop2 = Object.pop();

ObjectForPop2.Display();

return 0;

}



program 05 Class Template For Student Class only

// ConsoleApplication7.cpp : Defines the entry point for the console application.

//

#include "stdafx.h"

#include<iostream>

#include<string>

using namespace std;

class Student

{

int RollNo;

string Name;

float Cgpa;

public:

void input()

{

cout << "Etner the Roll No of Student : ";

cin >> RollNo;

cout << "Enter the Name of Student: ";

cin >> Name;

cout << "Enter the Cgpa of Student: ";

cin >> Cgpa;

}

void Display()

{

cout << "Roll NO of Student:" << RollNo << endl;

cout << "Name of Student: " << Name << endl;

cout << "Cgpa of Studnet: " << Cgpa << endl;

}

};

template <class TemplateOfStudentClass>

class Stack

{

TemplateOfStudentClass Array[5];

int top;

public:

Stack()

{

top = -1;

}

void Push(TemplateOfStudentClass GetValue)

{

if (top < 4)

Array[++top] = GetValue;

else

cout << "Stak is full\n";

}

TemplateOfStudentClass pop()

{

if (top > -1)

return Array[top--];

else

cout << "Stak is empty\n";

}

};

int main()

{

Student StudentObject, StudentObject2;

//StudentObject.input();

Stack<Student> Object;

StudentObject.input();

StudentObject2.input();

Object.Push(StudentObject);

Object.Push(StudentObject2);

cout << "\nNow Delete Last Value \n\n";

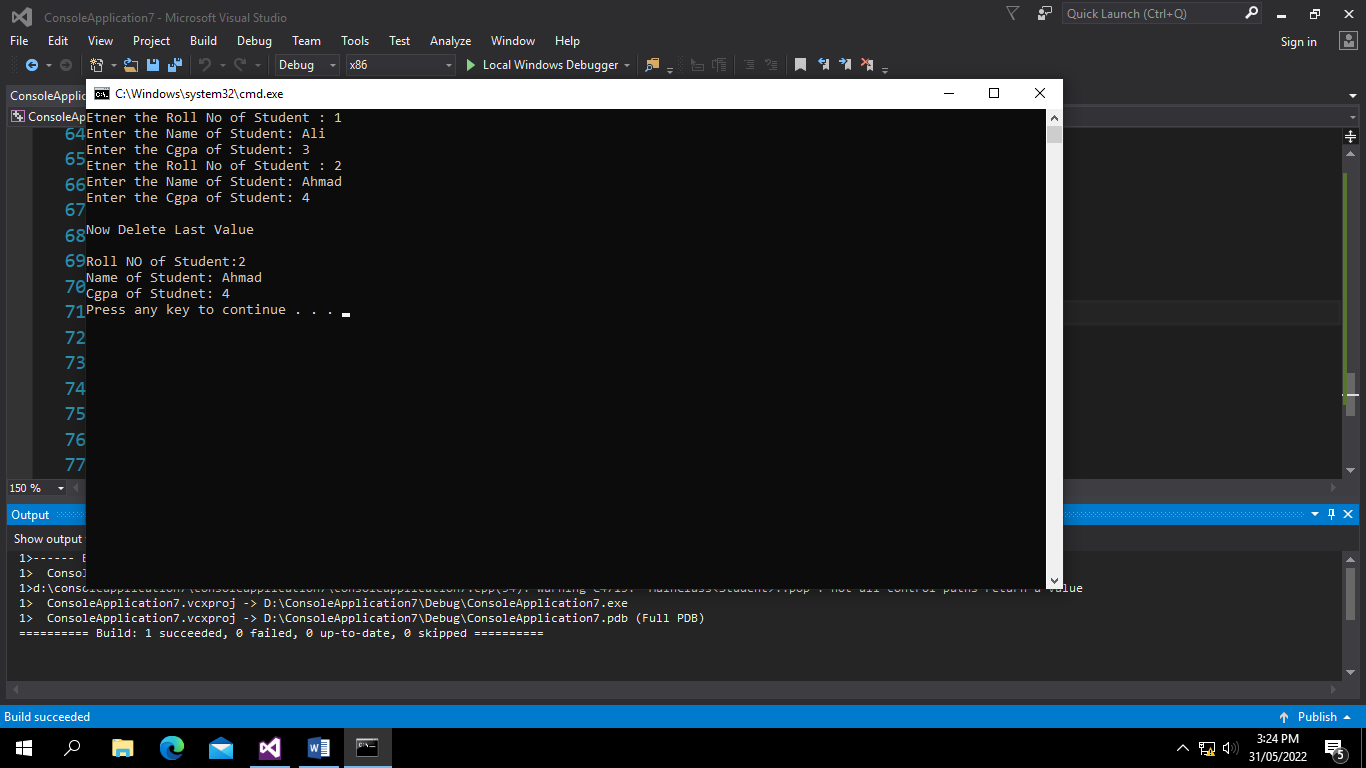
Student ObjectForPop, ObjectForPop2;

ObjectForPop = Object.pop();

ObjectForPop.Display();

return 0;

}



PROGRAM 05

TEMPLATE FOR Multiple classes

// ConsoleApplication7.cpp : Defines the entry point for the console application.

//

#include "stdafx.h"

#include<iostream>

#include<string>

using namespace std;

class Student

{

int RollNo;

string Name;

float Cgpa;

public:

void input()

{

cout << "Etner the Roll No of Student : ";

cin >> RollNo;

cout << "Enter the Name of Student: ";

cin >> Name;

cout << "Enter the Cgpa of Student: ";

cin >> Cgpa;

}

void Display()

{

cout << "Roll NO of Student:" << RollNo << endl;

cout << "Name of Student: " << Name << endl;

cout << "Cgpa of Studnet: " << Cgpa << endl;

}

};

template <class TemplateOfStudentClass>

class Stack

{

TemplateOfStudentClass Array[5];

int top;

public:

Stack()

{

top = -1;

}

void Push(TemplateOfStudentClass GetValue)

{

if (top < 4)

Array[++top] = GetValue;

else

cout << "Stak is full\n";

}

TemplateOfStudentClass pop()

{

if (top > -1)

return Array[top--];

else

cout << "Stak is empty\n";

}

};

class StackIntegers

{

int A[5];

public:

void Input()

{

cout << "Enter the Three Values for Integers: \n";

cin >> A[0] >> A[1] >> A[2];

}

void Display()

{

cout << "Your values are " << A[0] << " " << A[1] << " " << A[2] << " " << endl;

}

};

int main()

{

Student StudentObject, StudentObject2;

StackIntegers IntegerObject1, IntegerObject2;

Stack<Student> Object;

Stack<StackIntegers> Objectint;

StudentObject.input();

StudentObject2.input();

IntegerObject1.Input();

IntegerObject2.Input();

Object.Push(StudentObject);

Object.Push(StudentObject2);

cout << "Enter the Values of Integers\n";

Objectint.Push(IntegerObject1);

Objectint.Push(IntegerObject2);

cout << "\nNow Delete Last Value of students \n\n";

Student ObjectForPop;

ObjectForPop = Object.pop();

ObjectForPop.Display();

cout << "\nNow Delete Last Value of Integers \n\n";

StackIntegers IntegerForPop;

IntegerForPop = Objectint.pop();

IntegerForPop.Display();

return 0;

}

