A
PROJECT SYNOPSIS
ON

"Syntax Master"

Submitted to

Rashtrasant Tukadoji Maharaj Nagpur University,
NAGPUR
In the Partial Fulfillment of

B.Com. (Computer Application) Final Year

Synopsis Submitted by
Ananya Ghanshyam Srivastava
Aniket Narhari Deshmukh

Under the Guidance of

Pravin J. Yadao

G. S. College of Commerce & Economics
Nagpur

2019-2020

G.S COLLEGE OF COMMERCE & ECONOMICS

NAGPUR

CERTIFICATE

(2019-2020)

This is to certify that Miss Ananya Srivastava and Mr. Aniket Narhari Deshmukh has completed their project on the topic of "SYNTAX MASTER" prescribed by the Rashtrasant Tukadoji Maharaj Nagpur University for B.Com (Computer Application) – III course in G.S College of Commerce & Economics Nagpur.

Date-
Place- Nagpur

Pravin J. Yadao

Project Guide

External Examiner **Internal Examiner**

ACKNOWLEDGEMENT

We take the opportunity to express our deep gratitude and whole hearted thanks to project guide Prof. Pravin Yadao, Coordinator for his guidance throughout this work. We are very much thankful to him for his constant encouragement, support and kindness .

We are also grateful to our teachers Prof. Rahul Tiwari, Prof. Sushma Gawande , Prof. Preeti Rangari, Prof. Prajakta Deshpande and Prof. Haresh Naringe for their encouragement, help and support from time to time.

We also wish to express our sincere thanks to Principal Dr. N.Y Khandait for providing us wide range of opportunities, facilities and inspiration to gather professional knowledge and material without which this project could not have been completed.

Ananya Srivastava

Aniket Deshmukh

Date-

Place-Nagpur

DECLARATION

I **Mr. Aniket Narhari Deshmukh** and **miss Ananya Srivastava** hereby declare honestly declare that the work entitled **"SYNTAX MASTER"** submitted by us at G.S College of Commerce & Economics ,Nagpur in partial fulfillment of requirement for the award of B.Com(Computer Application) degree by Rashtrasant Tukadoji Maharaj Nagpur University, Nagpur has not been submitted elsewhere for the award of any degree ,during the academic session 2019-2020.

The project has been developed and completed by us independently under the supervision of the subject teacher and project guide.

Ananya Srivastava

Aniket Deshmukh

Date-

Place-Nagpur

INDEX

INTRODUCTION

INTRODUCTION

This project **"SYNTAX MASTER"** is basically a software which is developed for the student as well all learners. The basic motive of this project is to provide all types of syntax and example of all the basic programming languages which an IT person should know, so it is an information providing application. As this project contains the use of graphical user interface so it will be easy for the user to access and it's a very flexible software to use. As user can access to different forms of different programming languages very easily and can seek the information and learn from it.

These days Programming Languages are in very much boom state. Every next person is learning programming languages as they are in very much demand, as we all know day by day our technology is upgrading very rapidly and the main contribution in our country is of IT sector and this application helps in contribution to the education section for an IT student.

This project is basically divided into two parts one part for user only which is available in read mode only in which they can access the information and see the particular syntaxes and example. No user can alter the information which is provided in this section. In this part a user have to register and then they can access the particular information about the languages by logging in and it also consist of a feedback form in which user can give their feedback about this software.

And the second part of the project that is the admin part which is only accessible to the developer of this project, it will have a unique ID and password for the admin by which they can only access in admin section. So basically it will only generate the reports of the database that how many users have accessed to this software and the feedback given by the user.
The main motive behind the project is to provide all the syntax and example of the programming language and in one single application and so that user can have access to different sort of programming languages and there syntax , definitions , example and there outputs too under the same roof .

For using this software a user only needs the basic knowledge of Computer system and the valid user ID and Password which he/she has entered at the time of registration.

- This software is the GUI application which uses various controls and icons for making the software more user friendly.
- Reports can only be seen by the admin in very easy and convenient manner.
- By using this software users can have access to different programming languages whenever and wherever they want to.
- This is a completely offline application, it can run on any system and it does not require any internet to run.
- This system is secured because only the users who have registered can login into the application by their id and password.
- In this project all the information is connected from database and all the information is retrieving through the database, in this we have used MS Access as the backend.
- The reliability and durability can be seen by using the project and features various times. This software can be used at anywhere and at any platform.
- In future we will try to add more programming languages and more information about them and try to add many different examples of a particular Syntax.
- It can also be converted into a mobile based application.

OBJECTIVES

OBJECTIVES

The main objectives of **"SYNTAX MASTER "** are as follows:

1.User-Friendly-
It is Important that the software which is developed should user-friendly, So the system developed is user friendly and easily understandable by any user.

2.Easy to access-
As in our project MDI form has been taken so it is to access every form through the MDI form on one click.

3.Reduction in paper work-
This software reduces the paper work as all the syntax are available in this software no need of paper use all the things are available in the software only.

4.Time saving-
This software's objective is to save the time because different languages are available in only one software , so this automatically saves the time of the user.

5.Reliability-
This system can be used again and again .it has reliability feature by this we can use this application anywhere at any time and there is no time limit applicable for the use.

6.Secured access-
As in this software every user has his /her unique ID and Password or they can register also to create new ID or Password ,By this only those have registered can have access to the Software.

7.Completely Offline-
This software is completely Offline , no use of internet is required for using this software, So anyone can easily use this software without any internet.

8.Portability-

As this system is developed for different users , so this software is highly portable ,as one can use this project at any platform without installing any specialized software.

9.Cost Reduction-

As the system is fully automatic and it doesn't require any sophisticated software to get used and it also reduces the paper work required because all the work is done through computer so it reduces the cost .As this is the offline application so there is no need of spending money in internet connection.

10.Interactive-

As this system is developed by using Graphical User Interface (GUI).Hence this system contains icons and symbols which makes this system Interactive and easy to use.

11.Easy to handle-

This system is the combination of the important thing which is sometimes necessary for the users to access in very short time span.This software contains all such factors under one roof , so this system is easy to handle for the user.

12.Low Maintenance-

In this software SYNTAX MASTER there is very less need of Maintenance because this project is completely offline project and there Is very less chances of technical breakdowns in it.

13.Better Utilization of Resources-

Today user is more busy and want faster and time saving system **"SYNTAX MASTER "** can fulfill all the requirements as by enabling quick access of multiple option from a single screen , so that people can save their time by using this software.

PRELIMINARY SYSTEM ANALYSIS

PRELIMINARY SYSTEM ANALYSIS

PRELIMINARY INVESTIGATION-

Preliminary investigation is the initial investigation before the commencement of the software development.
In our project "SYNTAX MASTER" we first examined that the time consumption is too high because multiple software and
Websites are used for generating the results and seeing the information and declaring the reports and many other administrative things.
In this software we have just taken the criteria for the users which can have access for the information, a user can seek information about many programming languages under one roof. We just kept in mind that syntax of different programming languages can be available in a website but seeing it as an application is very rare, we can see different application of a particular programming language but if we want to search for an application which has different programming languages it is not available there, so main intention for making this project is that to provide information of many different programming languages in one application , it is a very useful application for an IT student.

From saving the information from unauthorized access, every user has to register in this application for creating their unique ID and Password.
This application can be used for exam perspective for students as well as for teachers.
For logging in the admin part of the application the admin will have its unique ID and Password which will be only accessible for the admin only. By this admin can seek all the reports which he wants to.

We just tried to implement this system for the day to day working in an educational field basically in the IT world. Thus in order to carryout project successfully it is to be taken under consideration that the user should provide the system with such things. This system should be user friendly and at the same time perform all the activities mentioned above with the speed and accuracy as this is essential in Current Scenario.

PRESENT SYSTEM IN USE

The current system which is being used in the institutes is very obsolete which requires updating on daily basis because there is very high number of programming languages. The system used in current system has only information and syntaxes about a particular language only in an application which is not that useful for an IT student.

Currently the present system does not have that much facility to have information of multiple programming languages we just have to access different application for different programming languages, so by using this app users can get different programming languages under one application they can see it 24x7, whenever they want to.

In the traditional system only information, syntaxes and examples of only one programming language is available but in our software multiple programming languages has been provided and their syntax with examples and outputs too are there in our software.

All the data has been called from the database and no user can manipulate the data, it is only available in Read Only mode. So that's why all the information can only be changed by the developer and not by the user and it is secured too in our software.

FLAWS IN THE PRESENT SYSTEM

1)Time Consuming-The current system in use is too much time consuming because all the information is are available in different platforms. But by using this application software one can save the time required for seeking different types of information, they can seek the information whenever and wherever they want.

2) Insecurity-The current system in use is not that much secure because any one can have access in the application , but in this application an user have to generate particular user ID and password to login in the application .

3)Rigidity-Present system in use is rigid so the required updations take too much time. As this system is fully automated and one can seek information easily and the reports can only be seen by the admin.

4)Requires Paper Work- The present system requires paper work as different syntaxes and example have been given the different books , so by this it uses lots of paper, but in our application all the information is just available in the application only.

5)Poor Performance Management- The present system has poor performance management as it only has the capability of generating information of only one programming language. As in our application multiple number of languages are available.

6)Less Chances of Updations- As in the present system there is less chances of updations as there is only information about a particular language so multiple languages cannot be added into it, but we can have updated things of different languages.

7)Resistant to Change-The current system in use is resistant to change as it requires more and more Hard work for implementing the updations which is required in the current scenario. As mentioned earlier this system has capability to update itself according to the need of the user hence this system will be compatible in today's changing environment.

NEED OF NEW SYSTEM

The current system on which we are working is semi-automatic i.e. in current system there are more than one platforms are required for performing the tasks and generating reports, Which requires more time for processing the input and generating the output in the current scenario the time is precious and we require the fully automatic system by which we can generate the reports and by which we can access the information whenever and wherever we want to.
In the current system the time required is very high and it is becoming obsolete day by day .To avoid these problems we need to make changes in the present system by developing the new system that satisfy all needs of the user, save time and provide quality result.

Following are the needs of the new System:-

1. Increased Operational Speed-
Today most of the user wants faster and user friendly applications and present system does not meet such requirement it leads to development of new faster application system which fulfills all the requirement of the user in shorter time interval. This system is totally computer based which is obviously increase the operational speed automatically.

2.Cost Reductions-
The current system in use requires huge amount of paper work which increases the cost as well as harms the environment. But by applying the new system the cost involved will decrease and this will also save the environment because there is negligible use of paper is required.

3.Increased degree of Transparency-
In the current system there is just a one way communication which has very low transparency but by applying the new system in which the teachers as well as students are connected to this software and clear their doubts according to their e-requirements from the concerned page.

4.Interactive-

The current system is not too much interactive but the user nowadays wants the interactive system which can easily be used. In this system we have used various icons and Graphical User Interface (GUI) for making the software user friendly and interactive.

5.Easy To Manage-

Managing the current system is very tedious task because there are various platforms are used but in the system all the works are performed under one platform so this project is easy to manage and analysis can be done easily.

6.Ease in information seeking-

As this software is developed for both students as well as teachers and other users also , as teachers can have the examples and use in their teaching and students can have information and use for their learning and exams.

7.Faster Performance-

As this software is totally computerized and it is compatible with the computers which contain the moderate specifications so the performance of the newly developed system is faster as compared to the older system.

FEASIBILITY STUDY

Feasibility study of a system means whether the system is practically possible to build or not. It also evaluates the benefits of the new system. In "Multi-Utility Software" we are not able to solve all the types of errors but to find out the various type of solution and pick out the best one as relates to our software. The feasibility study means to analyze the requirements to the user. Project manager use feasibility studies to determine potential positive and negative outcomes of a project before investing a consideration amount of time and money into it.

In **"Syntax Master"** feasibility study considered three major keys, they are-
1)Technical Feasibility
2) Economical Feasibility
3)Operational /Behavioral Feasibility

Technical Feasibility-

Technical Feasibility is one of the first studies that must be conducted after a project has been identified. Technical Feasibility centers on the existing computer system that is the availability of the required hardware , software and operating system. This project is developed in Microsoft Visual Studio 2010, which can be easily copied and run on any system with the required configuration. Technical feasibility means to solve the problem as related to the hardware and software. It refers to the technical resources needed to develop a new system. The analyst must find out whether current technologies are sufficient to proposed system which includes that in "Syntax Master" the problems like showing the information of the selected topic can occur. The analyst must find out that the current system and technology is able to tackle the situation and solve the problem or not and the developed system is considered technically feasible if the internal technical capability is sufficient to protect the system.

Economic Feasibility-

Economic Feasibility determines whether the required software is capable of generating financial gains for an organization. It involves the cost incurred on the software development team, estimated cost of hardware and software , cost of performing feasibility study, and so on.

For this, it is essential to consider expenses made on purchase (such as hardware purchase) and activities required to carry out software development.

In addition, it is necessary to consider the benefits that can be achieved by developing the software. Software is said to be economically feasible if it focuses on the issues listed below-

- Cost incurred on the software development to produce long - term gains for an organization.
- Cost required to conduct full software investigation (such as requirements elicitation and requirements analysis)
- Cost of hardware , software and development team and training.

This software is fulfilling all the three points mentioned above because this software will cut down the use of papers and this will be very cost saving for the organization and this software doesn't require any specialized platform and any specialized to work so there is no need for any customer to update the system just for using this software , this software contains GUI and it is user friendly as well so there is no need for any specialized person to operate this software so it also reduces the cost involved in training etc.

Operational Feasibility-

Operational feasibility refers to the measure of solving problems with the help of a new proposed system. It helps in taking advantage of the opportunities and fulfills the requirements as identified during the development of the project. It takes care that the management and the users support the project.Operational feasibility is the measure of how well a proposed system solves the problems , and takes advantage of the opportunities identified during scope definition and how it satisfies the requirements identified in the requirements analysis phase of the system development. The operational feasibility assessment focuses on the degree to which the proposed

development project fits in with the existing business environment and objectives with regard to development and existing business processes.
To ensure success , desired operational outcomes must be imparted during design and development. These include such design – dependent parameters such as reliability , maintainability, supportability , usability , productivity, disposability, sustainability, affordability and others. These parameters are required to be considered at the early stages of design if desired operational behaviors are to be realized. A system design and development requires appropriate and timely application of engineering and management efforts to meet the previously mentioned parameters. A system may serve its intended purpose most effectively when its technical and operating characteristics are engineered into the design. Therefore , operational feasibility is a critical aspect of systems engineering that needs to be an integral part of the early design phases.

The software "SYNTAX MASTER" is supposed that it solves the problem of the current system. As mentioned earlier that this system will eradicate the use of the papers in the actual ground also the project will play the vital role in educational field. This will also reduce cost in actual ground and fulfill each and every objective written above.

PROJECT CATEGORY

PROJECT CATEGORY

PLATFORM/LANGUAGES/TOOLS USED:
FRONTEND: VISUAL BASIC. NET
BACKENED: MS ACCESS

This project "SYNTAX MASTER" uses Microsoft VB.Net as frontend and Microsoft Access as backend of this project. Microsoft Windows 10 as platform to know working of the project one should know about its platform. Brief review of the package-

Microsoft Visual Basic.Net:

Visual basic.Net(VB.NET) is an object-oriented computer programming language implemented on the .Net Framework. Although it is an evolution of classic Visual Basic language, it is not backwards- compatible with VB6, and any code written in the old version does not compile under VB.Net.
Like all other .Net languages, VB.Net has complete support for Object-oriented concepts. Everything in VB.Net is an object, including all of the primitive types (Short, Integer, Long, Boolean, etc) and user- defined types , events and even assemblies. All objects inherits from the base class object.

VB.Net is implemented by Microsoft's .Net framework. Therefore , it has full access to all the libraries in the .Net framework. It's also possible to run VB.Net programs on mono, the open source alternative to .Net, not only under the windows, but even Linux or Mac OSX.

The following reasons make VB.Net a widely used professional language-

- Modern, general purpose
- Object oriented
- Component oriented
- Easy to learn
- Structured language
- It produces efficient programs
- It can be compiled on a variety of computer platforms
- Part of .Net Framework

MS ACCESS:

Microsoft Access is a database management system (DBMS) from Microsoft that combines the relational Microsoft Jet Database Engine with a graphical user interface and software development tools. It is a member of the Microsoft Office suite of applications, included in the professional and higher editions or sold separately.

Microsoft Access stores data in its own format based on the access Jet Database Engine. It can also import or link directly to data stored in other applications and databases.

Software developers, data architects and power users can use Microsoft Access to develop application software. Like other Microsoft Office applications , access is supported by Visual Basic for Applications (VBA), an object- base programming language that can reference a variety of objects including DAO(Data Access Objects), ActiveX Data Objects, and many other ActiveX components , Visual objects used in forms and reports expose their methods and properties in the VBA programming environment , and VBA code modules may declare and call Windows operating system operations.

HARDWARE AND SOFTWARE REQUIREMENT SPECIFICATIONS

HARDWARE AND SOFTWARE REQUIREMENT SPECIFICATIONS

HARDWARE-

This software is prepared keeping in mind the requirement of the user. To run this software properly without any problem we need to fulfill the following minimum hardware requires.

- **Hardware requirements**

Processor : Intel Core 13
RAM : 4 GB
Hard Disk : 500 GB or more
Monitor : VGA
Input: Keyboard and mouse

SOFTWARE-

Software can be termed as the group of instructions or command used by the computer to accomplish the given task. In today's world generation of software is ever ending. It is an evolution of dignified technology. There is no requirement for it but the minimum software requirement of any computer is specified below:-

- **Software requirements**

Operating System –Microsoft Windows based operating system like Windows 7, Windows 8 or Windows 10.
Frontend - Microsoft Visual Studio 2010
Backend - MS Access

Frontend-

VB.Net is based on the classic Basic language for major of its syntax are similar to those in Basic language. It provides GUI(Graphical User Interface) for the development of applications. It supports Event Driven Programming. It provides common programming platform across all MS-Office applications. It is easy Error Handing and debugging facilities. It has a easy connectivity to Microsoft Access.

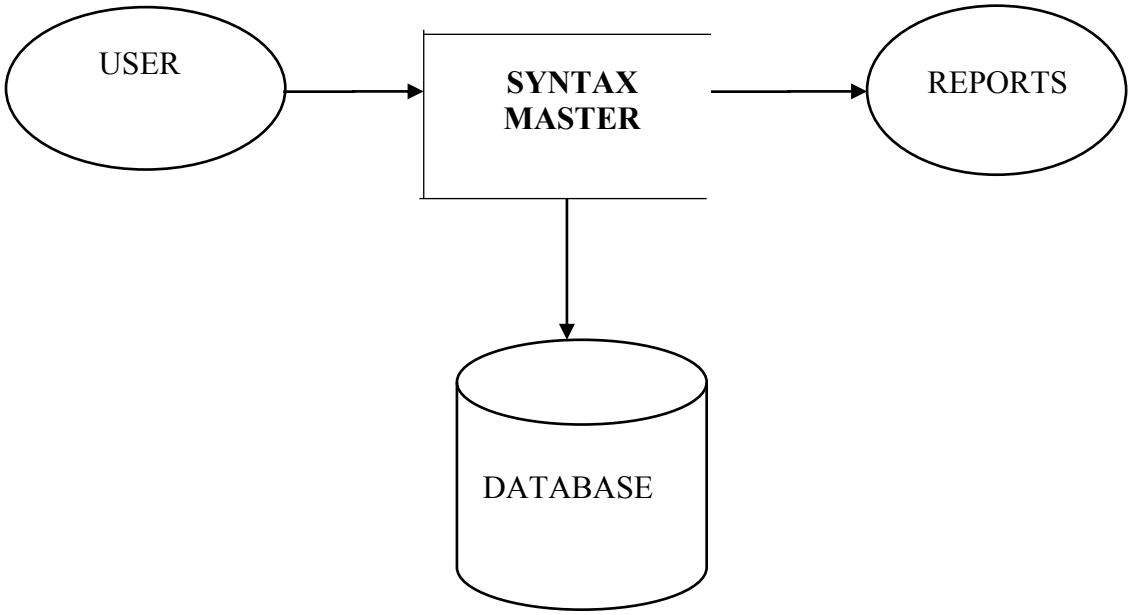
Backend-

Microsoft Access is a software package for database management. Access is an easy to use for DBMS. It is a full-featured database for the development of applications. It comes with various graphics, functions, wide range of query designers, reports, etc. No programming required to perform these operations. Data is stored in the form of tables.

DETAILED SYSTEM ANALYSIS

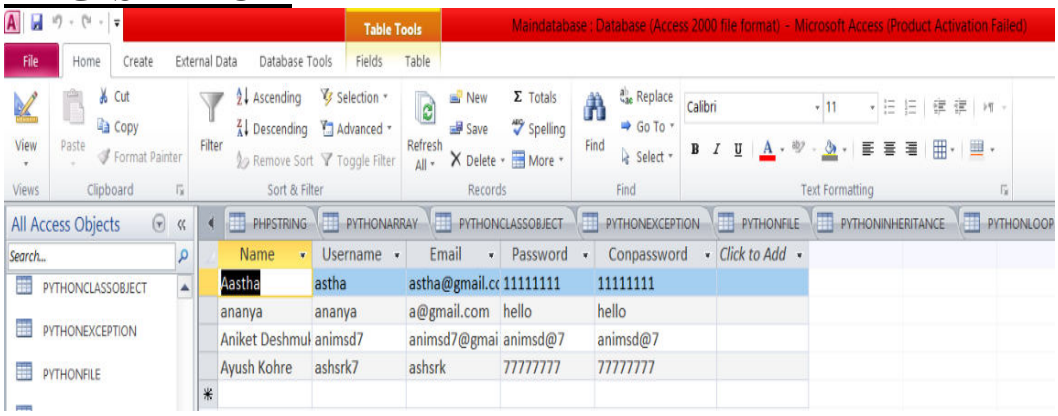
DETALED SYSTEM ANALYSIS

DATA FLOW DIGRAM:

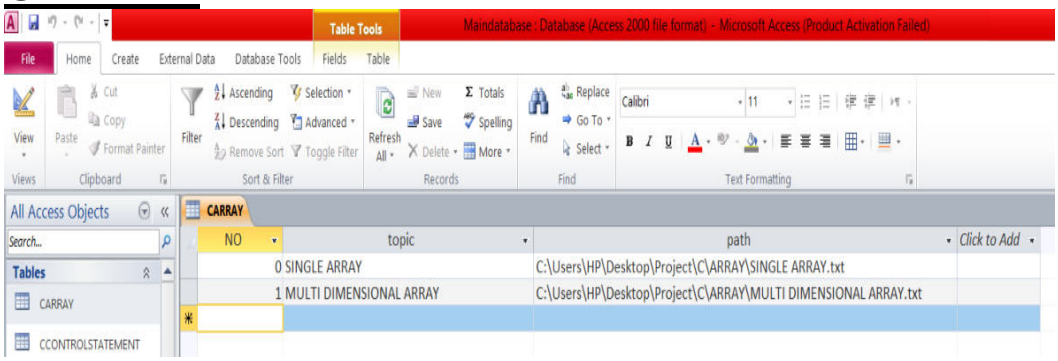


DATA STRUCTURE AND TABLES

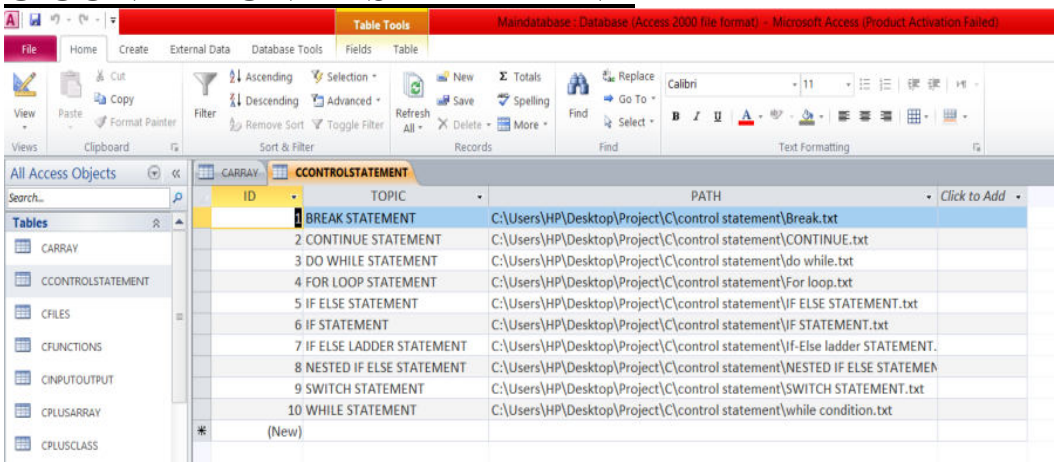
REGISTEROK



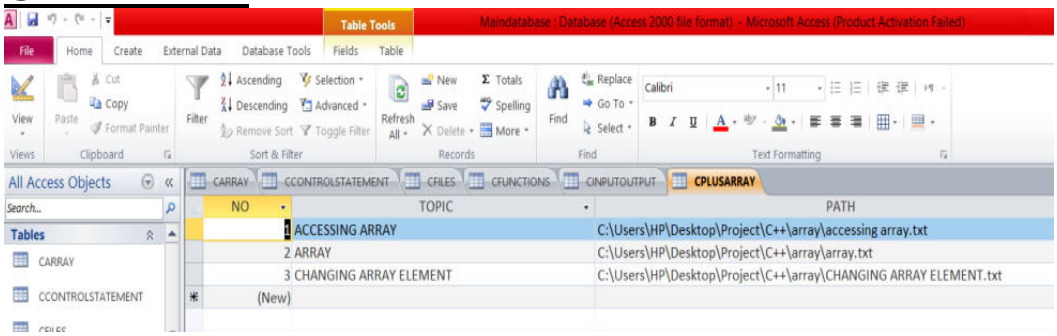
C ARRAY



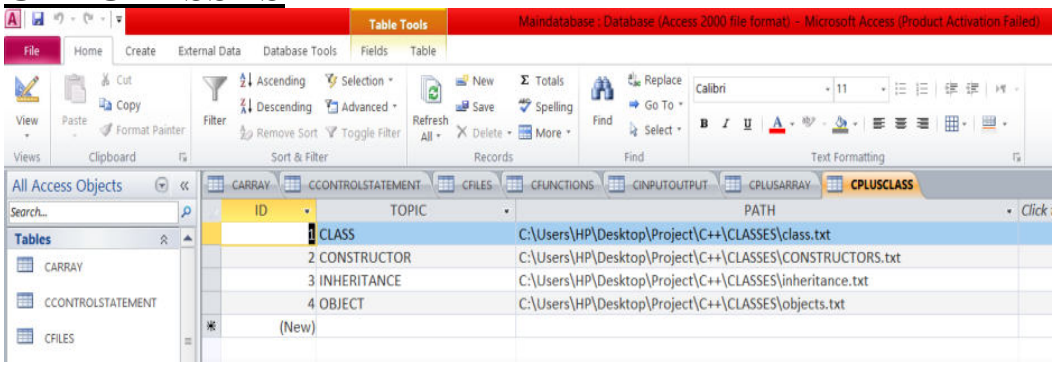
C CONDITIONAL STATEMENT



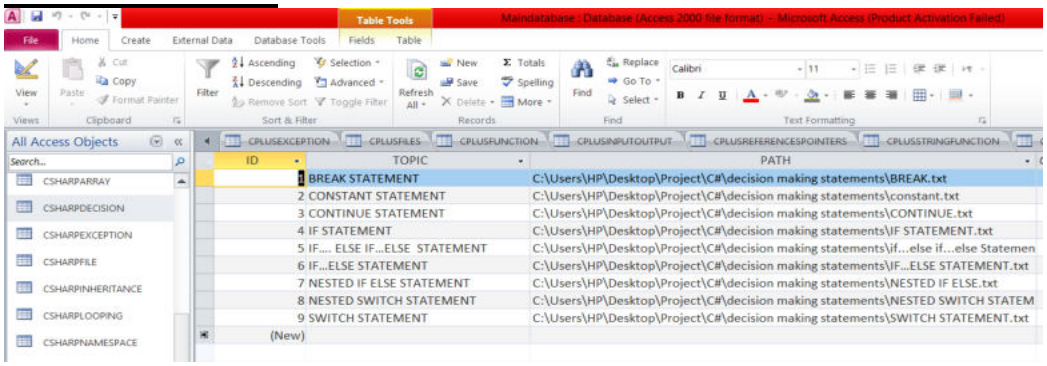
C++ ARRAY



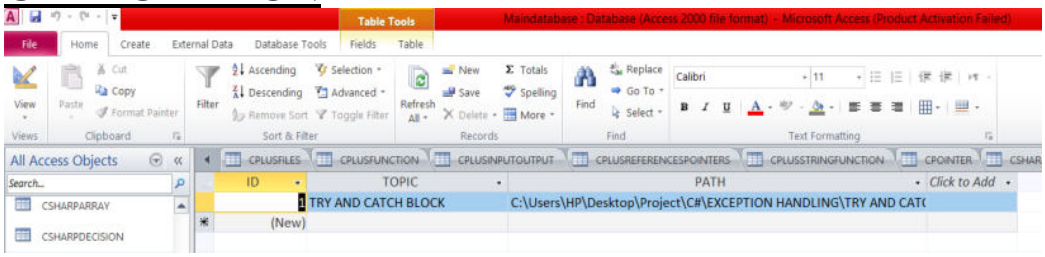
C++ CLASSES



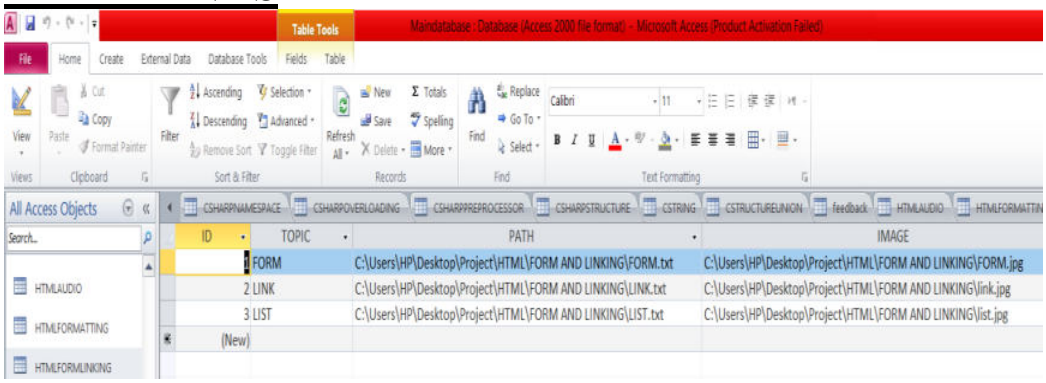
C# DECISION



C# EXCEPTION

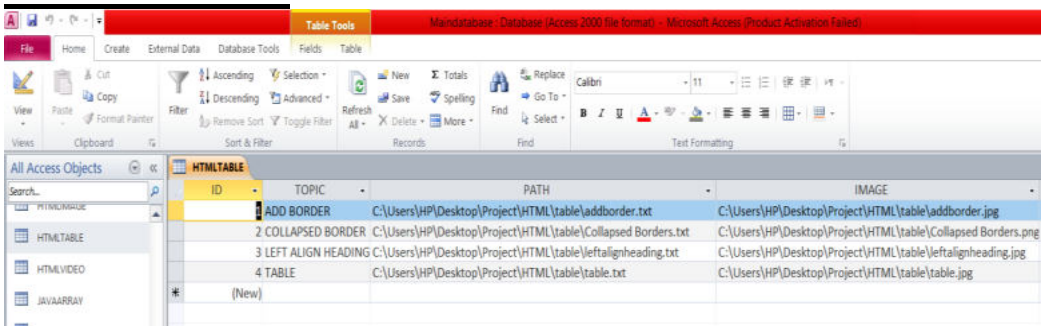


HTML LINKS



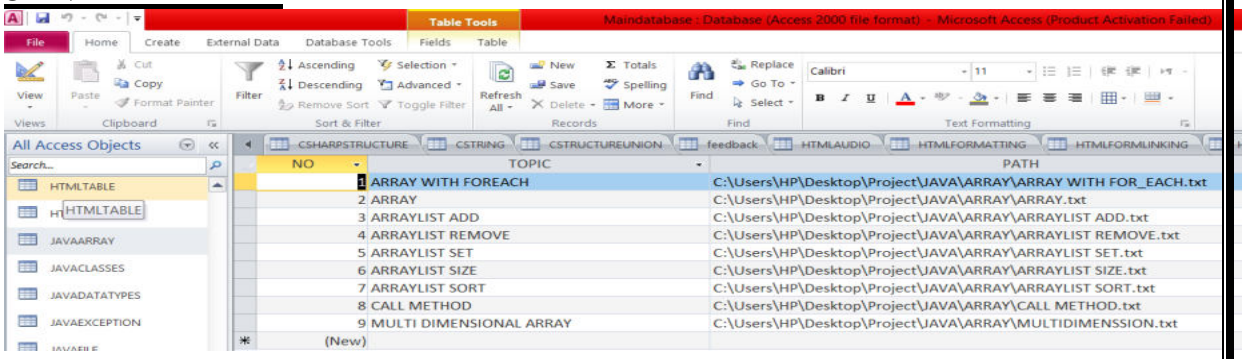
ID	TOPIC	PATH	IMAGE
1	FORM	C:\Users\HP\Desktop\Project\HTML\FORM AND LINKING\FORM.txt	C:\Users\HP\Desktop\Project\HTML\FORM AND LINKING\FORM.jpg
2	LINK	C:\Users\HP\Desktop\Project\HTML\FORM AND LINKING\LINK.txt	C:\Users\HP\Desktop\Project\HTML\FORM AND LINKING\link.jpg
3	LIST	C:\Users\HP\Desktop\Project\HTML\FORM AND LINKING\LIST.txt	C:\Users\HP\Desktop\Project\HTML\FORM AND LINKING\list.jpg
(New)			

HTML TABLE



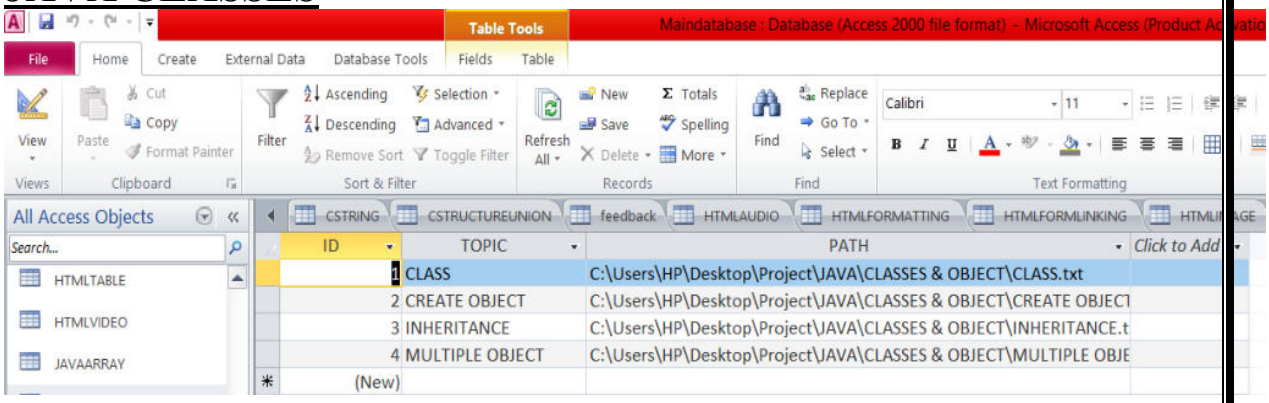
ID	TOPIC	PATH	IMAGE
1	ADD BORDER	C:\Users\HP\Desktop\Project\HTML\table\addborder.txt	C:\Users\HP\Desktop\Project\HTML\table\addborder.jpg
2	COLLAPSED BORDER	C:\Users\HP\Desktop\Project\HTML\table\Collapsed Borders.txt	C:\Users\HP\Desktop\Project\HTML\table\Collapsed Borders.png
3	LEFT ALIGN HEADING	C:\Users\HP\Desktop\Project\HTML\table\leftalignheading.txt	C:\Users\HP\Desktop\Project\HTML\table\leftalignheading.jpg
4	TABLE	C:\Users\HP\Desktop\Project\HTML\table\table.txt	C:\Users\HP\Desktop\Project\HTML\table\table.jpg
(New)			

JAVA ARRAY



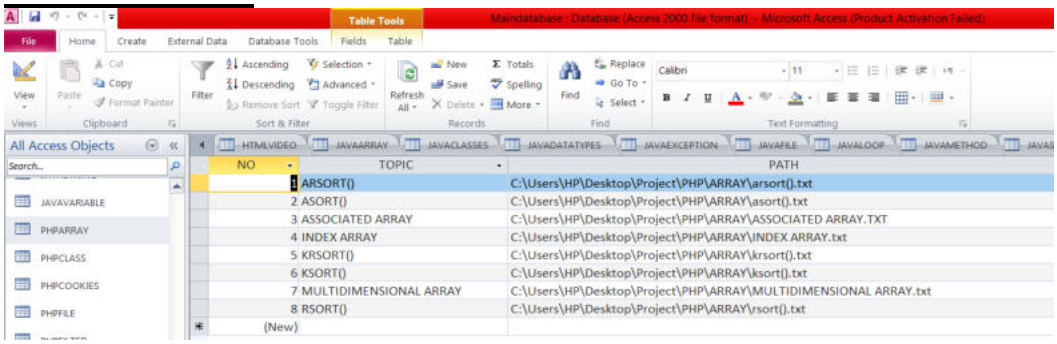
NO	TOPIC	PATH
1	ARRAY WITH FOREACH	C:\Users\HP\Desktop\Project\JAVA\ARRAY\ARRAY WITH FOR_EACH.txt
2	ARRAY	C:\Users\HP\Desktop\Project\JAVA\ARRAY\ARRAY.txt
3	ARRAYLIST ADD	C:\Users\HP\Desktop\Project\JAVA\ARRAY\ARRAYLIST ADD.txt
4	ARRAYLIST REMOVE	C:\Users\HP\Desktop\Project\JAVA\ARRAY\ARRAYLIST REMOVE.txt
5	ARRAYLIST SET	C:\Users\HP\Desktop\Project\JAVA\ARRAY\ARRAYLIST SET.txt
6	ARRAYLIST SIZE	C:\Users\HP\Desktop\Project\JAVA\ARRAY\ARRAYLIST SIZE.txt
7	ARRAYLIST SORT	C:\Users\HP\Desktop\Project\JAVA\ARRAY\ARRAYLIST SORT.txt
8	CALL METHOD	C:\Users\HP\Desktop\Project\JAVA\ARRAY\CALL METHOD.txt
9	MULTI DIMENSIONAL ARRAY	C:\Users\HP\Desktop\Project\JAVA\ARRAY\MULTIDIMENSSION.txt
(New)		

JAVA CLASSES



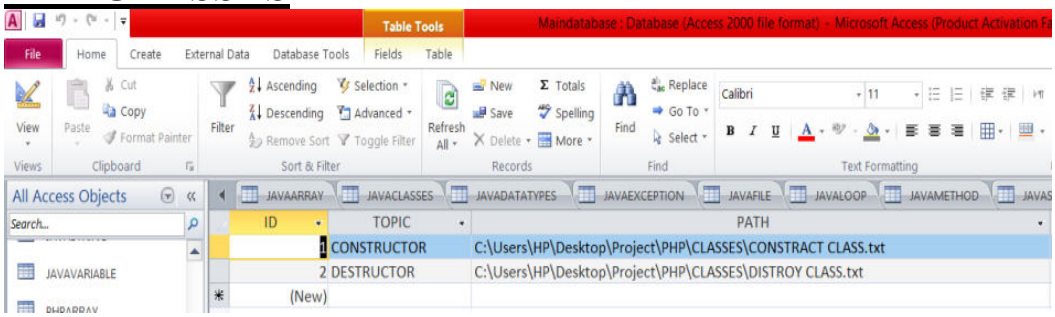
ID	TOPIC	PATH
1	CLASS	C:\Users\HP\Desktop\Project\JAVA\CLASSES & OBJECT\CLASS.txt
2	CREATE OBJECT	C:\Users\HP\Desktop\Project\JAVA\CLASSES & OBJECT\CREATE OBJECT
3	INHERITANCE	C:\Users\HP\Desktop\Project\JAVA\CLASSES & OBJECT\INHERITANCE.t
4	MULTIPLE OBJECT	C:\Users\HP\Desktop\Project\JAVA\CLASSES & OBJECT\MULTIPLE OBJE
(New)		

PHP ARRAY



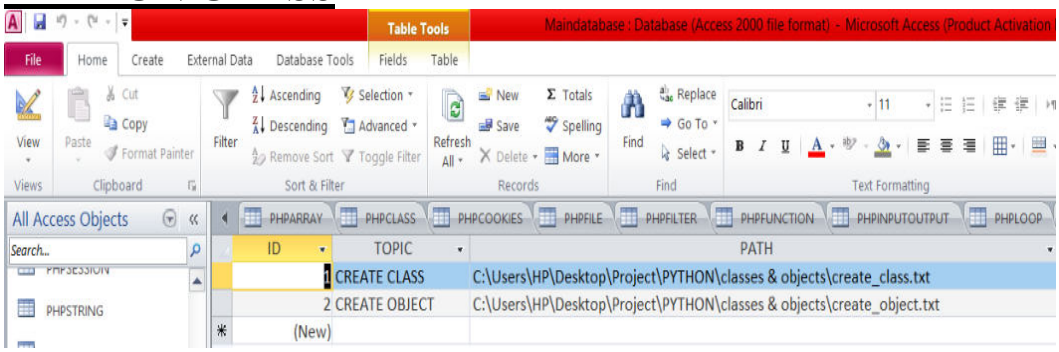
NO	TOPIC	PATH
1	ARSORT()	C:\Users\HP\Desktop\Project\PHP\ARRAY\arsort().txt
2	ASORT()	C:\Users\HP\Desktop\Project\PHP\ARRAY\asort().txt
3	ASSOCIATED ARRAY	C:\Users\HP\Desktop\Project\PHP\ARRAY\ASSOCIATED ARRAY.TXT
4	INDEX ARRAY	C:\Users\HP\Desktop\Project\PHP\ARRAY\INDEX ARRAY.txt
5	KRSORT()	C:\Users\HP\Desktop\Project\PHP\ARRAY\krsort().txt
6	KSORT()	C:\Users\HP\Desktop\Project\PHP\ARRAY\ksort().txt
7	MULTIDIMENSIONAL ARRAY	C:\Users\HP\Desktop\Project\PHP\ARRAY\MULTIDIMENSIONAL ARRAY.txt
8	RSORT()	C:\Users\HP\Desktop\Project\PHP\ARRAY\rsort().txt

PHP CLASSES



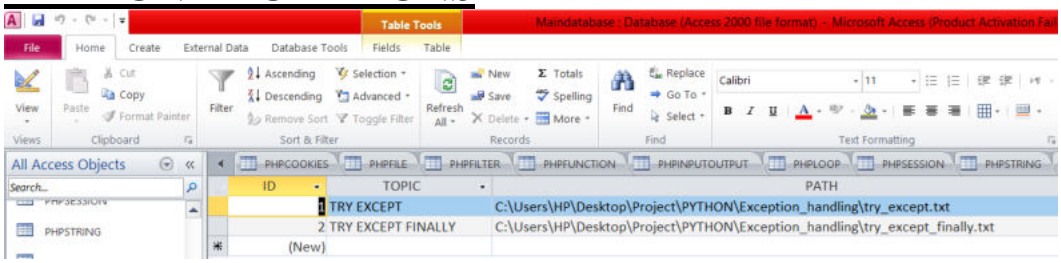
ID	TOPIC	PATH
1	CONSTRUCTOR	C:\Users\HP\Desktop\Project\PHP\CLASSES\CONSTRACT CLASS.txt
2	DESTRUCTOR	C:\Users\HP\Desktop\Project\PHP\CLASSES\DISTROY CLASS.txt

PYTHON CLASS



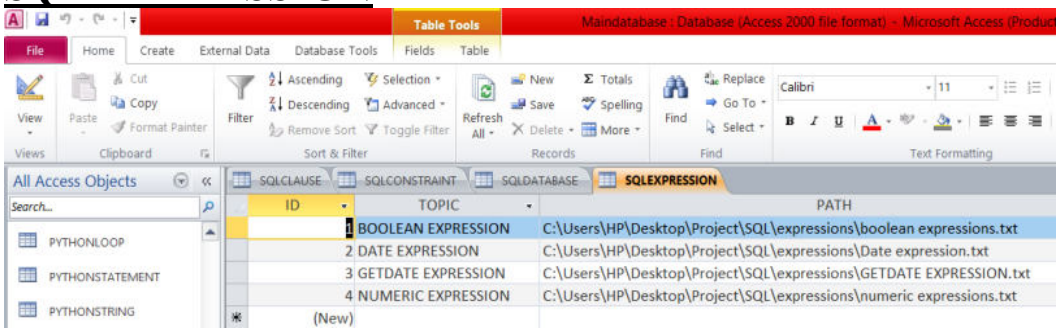
ID	TOPIC	PATH
1	CREATE CLASS	C:\Users\HP\Desktop\Project\PYTHON\classes & objects\create_class.txt
2	CREATE OBJECT	C:\Users\HP\Desktop\Project\PYTHON\classes & objects\create_object.txt

PYTHON EXCEPTIONS

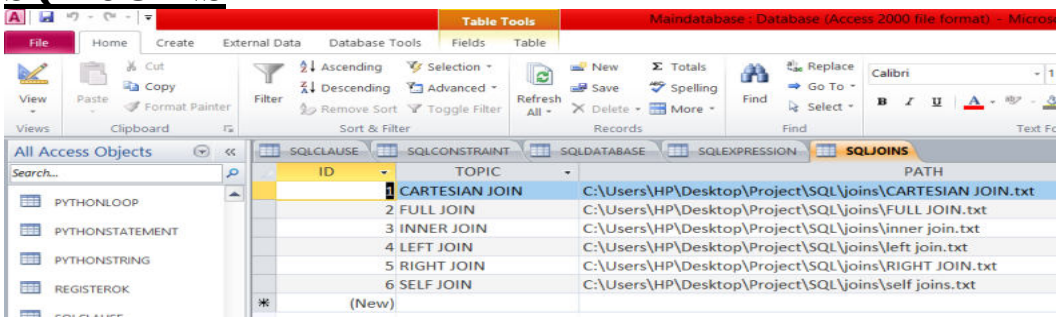


ID	TOPIC	PATH
1	TRY EXCEPT	C:\Users\HP\Desktop\Project\PYTHON\Exception_handling\try_except.txt
2	TRY EXCEPT FINALLY	C:\Users\HP\Desktop\Project\PYTHON\Exception_handling\try_except_finally.txt

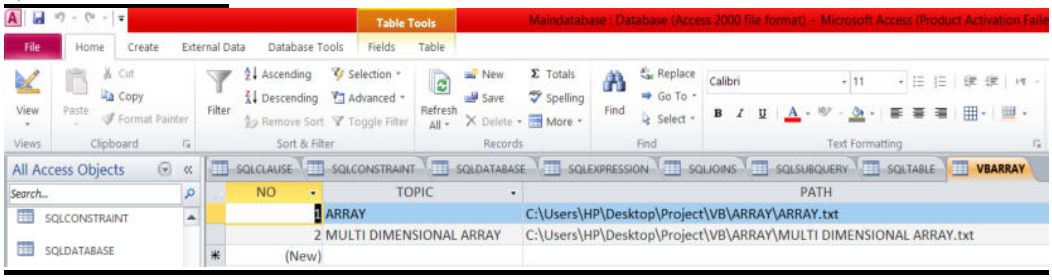
SQL EXPRESSION



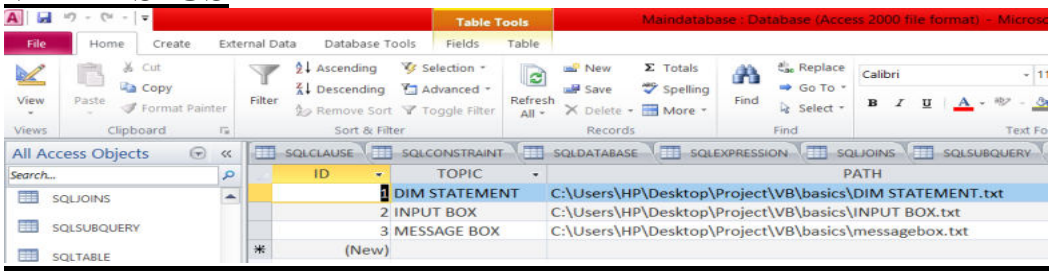
SQL JOINS



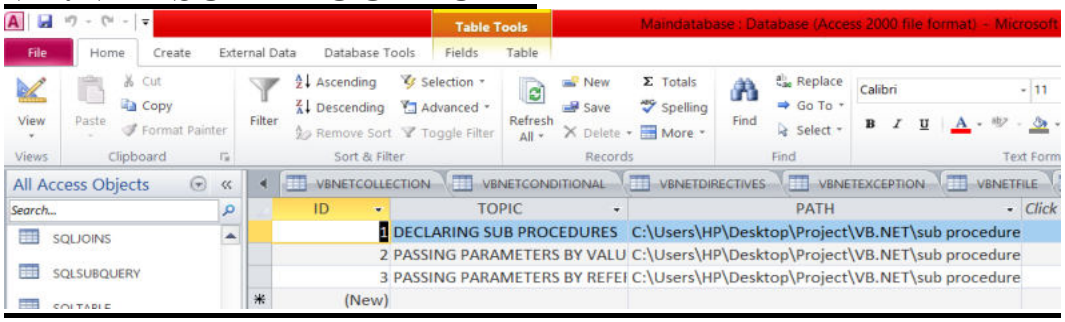
VB ARRAY



VB BASICS

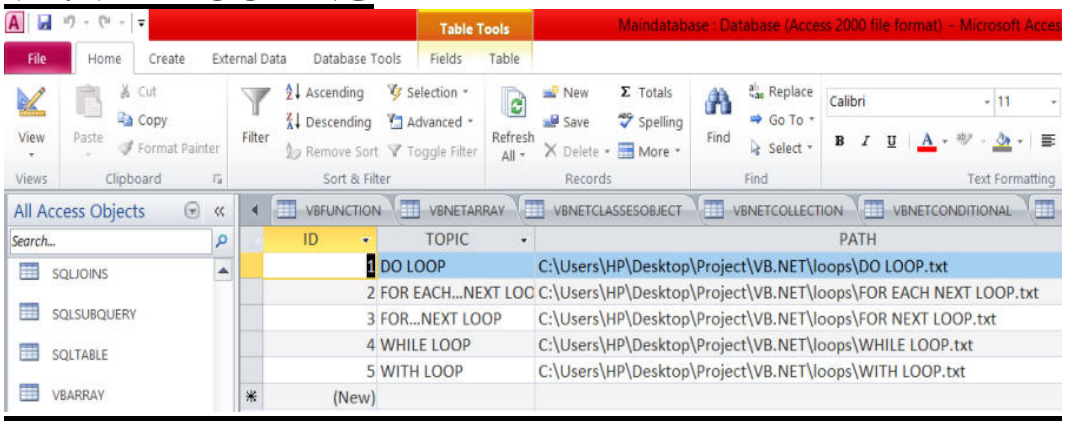


VB.NET SUB PROCEDURE



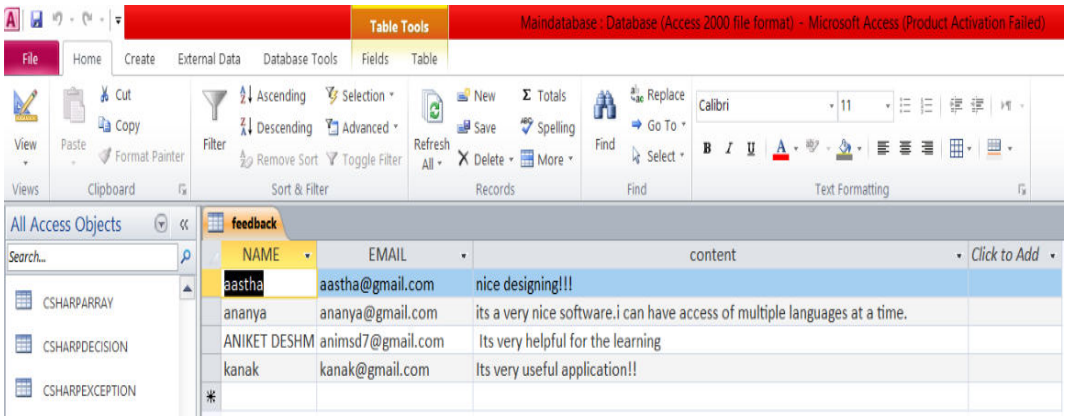
ID	TOPIC	PATH
1	DECLARING SUB PROCEDURES	C:\Users\HP\Desktop\Project\VB.NET\sub procedure
2	PASSING PARAMETERS BY VALU	C:\Users\HP\Desktop\Project\VB.NET\sub procedure
3	PASSING PARAMETERS BY REFEI	C:\Users\HP\Desktop\Project\VB.NET\sub procedure
(New)		

VB.NET LOOPING



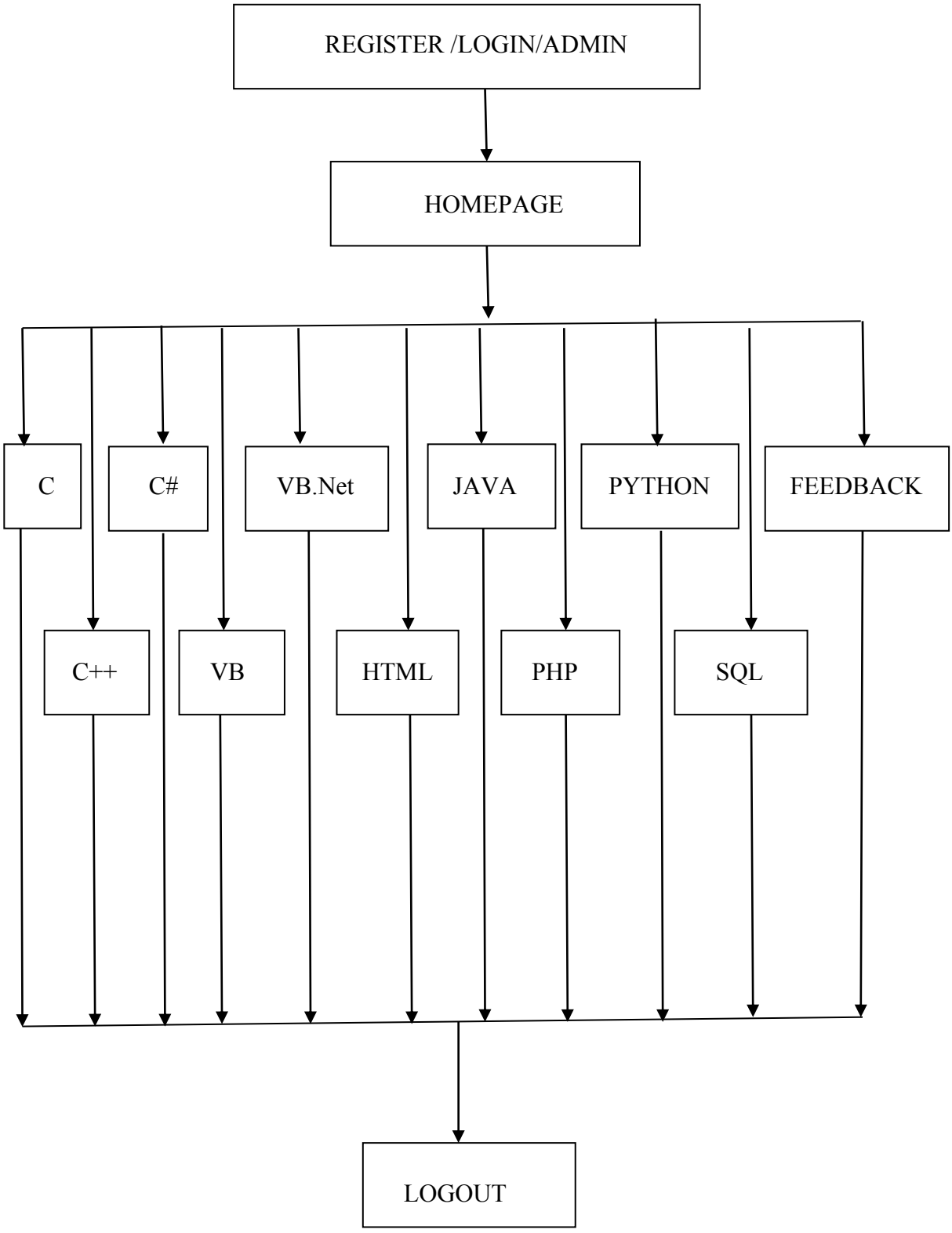
ID	TOPIC	PATH
1	DO LOOP	C:\Users\HP\Desktop\Project\VB.NET\loops\DO LOOP.txt
2	FOR EACH...NEXT LOO	C:\Users\HP\Desktop\Project\VB.NET\loops\FOR EACH NEXT LOOP.txt
3	FOR...NEXT LOOP	C:\Users\HP\Desktop\Project\VB.NET\loops\FOR NEXT LOOP.txt
4	WHILE LOOP	C:\Users\HP\Desktop\Project\VB.NET\loops\WHILE LOOP.txt
5	WITH LOOP	C:\Users\HP\Desktop\Project\VB.NET\loops\WITH LOOP.txt
(New)		

FEEDBACK



NAME	EMAIL	content
aastha	aastha@gmail.com	nice designing!!!
ananya	ananya@gmail.com	its a very nice software.i can have access of multiple languages at a time.
ANIKET DESHM	animsd7@gmail.com	Its very helpful for the learning
kanak	kanak@gmail.com	Its very useful application!!

STRUCTURE OF WINDOWS APPLICATION



SYSTEM DESIGN

SOURCE CODE

Registeration:

```
ImportsSystem.Data.OleDb
ImportsSystem.Data
ImportsSystem.Text.RegularExpressions
PublicClassRegistration
Public temp = 5
PrivateSub Button1_Click(ByVal sender AsSystem.Object, ByVal e As
System.EventArgs) Handles Button1.Click
Dimnama, uname, password, email, jkAsString
DimidusrAsInteger

nama = txtname.Text
uname = TextBox2.Text
password = TextBox3.Text
email = txtemail.Text

Iftxtname.Text=""Or TextBox2.Text=""Or TextBox3.Text=""Or
TextBox4.Text = ""Ortxtemail.Text = ""Then
MsgBox("Please Fill All The Box First !!!")
ElseIf TextBox3.Text <> TextBox4.Text Or TextBox3.TextLength <= 7 Then
MsgBox("Password do not match or missing !!!")
Else
Dim db source As String = "Provider=Microsoft.Jet.OLEDB.4.0
Source=C:\Users\HP\Documents\Registration.accdb"
Dim conn = NewOleDbConnection(dbsource)
Dimstr = "Insert into
[User]([IDUSR],[Nama],[Uname],[Pass],[JenisKelamin],[Email]) Values
('"&idusr&"','"&nama&"','"&uname&"','"& password &"','"&jk&"','"& email
&"') ;"
DimcmdAsOleDbCommand = NewOleDbCommand(str, conn)
Dim regex AsRegex = NewRegex("^([\w-\.]+)@((\[[0-9]{1,3}\.[0-9]{1,3}\.[0-
9]{1,3}\.)|(([\w-]+\.)+))([a-zA-Z]{2,4}|[0-9]{1,3})(\]?)$")
```

```
DimisvalidAsBoolean = regex.IsMatch(txtemail.Text.Trim)
Iftxtemail.Text = ""Then
MsgBox("Please enter Email!!")

ElseIfNotisvalidThen
'If isvalid<>txtemail.Text Then
MsgBox("Invalid Email Address")
Else
MsgBox("Ragistration Completed")

txtemail.Text = ""
txtname.Text = ""
            TextBox2.Text = ""
            TextBox3.Text = ""
            TextBox4.Text = ""
            ComboBox1.Text = ""
txtname.Focus()
Home.Show()
EndIf
Try
cmd.ExecuteNonQuery()
cmd.Dispose()
Catch ex AsException
'       MsgBox("Ragistration Completed")
EndTry
EndIf
EndSub

PrivateSub txtname_KeyPress1(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs) Handlestxtname.KeyPress
IfAsc(e.KeyChar) < 65 OrAsc(e.KeyChar) > 90 AndAsc(e.KeyChar) < 97
OrAsc(e.KeyChar) > 122 Then
e.Handled = True
MsgBox("You can only input letters!!")
EndIf
EndSub
```

```
PrivateSubtxtname_TextChanged(ByVal sender AsSystem.Object, ByVal e
AsSystem.EventArgs) Handlestxtname.TextChanged
Iftxtname.Text<>""Then
Dim a AsString = txtname.Text
txtname.Text = (StrConv(a, VbStrConv.ProperCase))
txtname.Select(txtname.Text.Length, 0)
EndIf
EndSub
PrivateSub Button2_Click(ByVal sender AsSystem.Object, ByVal e
AsSystem.EventArgs) Handles Button2.Click
End
EndSub
EndClass
```

LOGIN:-

```
Public Class Login
    Dim inc As Integer
    Dim maxrows As Integer
    Dim con As New OleDb.OleDbConnection
    Dim ds As New DataSet
    Dim da As New OleDb.OleDbDataAdapter
    Dim sql As String

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
Registration.Show()
    End Sub
 Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
maxrows = ds.Tables("AddressBook").Rows.Count
inc = -1
Home.Show()
    End Sub

    Private Sub BindingNavigator1_RefreshItems(ByVal sender As
System.Object, ByVal e As System.EventArgs)
```

```
Handles BindingNavigator1.RefreshItems
        TextBox1.Text = ds.Tables("AddressBook").Rows(inc).Item(1)
        TextBox2.Text = ds.Tables("AddressBook").Rows(inc).Item(2)
End Sub End Class
```

ADMIN LOGIN-

```
PublicClassADMIN
PrivateSubADMIN_Load(sender AsObject, e AsEventArgs)
HandlesMyBase.Load
Login.Hide()
EndSub
PrivateSubloginbtn_Click(sender AsObject, e AsEventArgs)
Handlesloginbtn.Click
Ifuser.Text = "admin"Andpassword.Text = "admin"Then
Form3.Show()
Else
MsgBox("Incorrect User ID or Password")
EndIf
EndSub
PrivateSub Button1_Click(sender AsObject, e AsEventArgs) Handles
Button1.Click
Login.Show()
EndSub
PrivateSub CheckBox1_CheckedChanged(sender AsObject, e AsEventArgs)
Handles CheckBox1.CheckedChanged
If CheckBox1.Checked = TrueThen
password.PasswordChar = ""
Else
password.PasswordChar = "*"
EndIf
EndSub EndClass
```

ADMIN HOME-

```
Public Class Form4
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
        Login.Show()
    End Sub
 Private Sub Form4_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        Form3.Hide()    End Sub
```

```
Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
        Me.Hide() REPORTS.Show()
    End Sub
  Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
        Me.Hide()
        report2.Show()
    End Sub
End Class
```

UPDATE PASSWORD-

```
ImportsSystem.Data.OleDb
PublicClassForm2
PrivateSub Button1_Click(ByVal sender AsSystem.Object, ByVal e
AsSystem.EventArgs) Handles Button1.Click
Try
DimsqlconnAsNewOleDbConnection
DimsqlqueryAsNewOleDbCommand
DimcmdAsNewOleDbCommand("SELECT Username FROM
REGISTEROK WHERE Username='"&user.Text&"'", sqlconn)
DimconnStringAsString
connString = "Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=C:\Users\admin\Desktop\Project\database\Maindatabase.mdb"
sqlconn.ConnectionString = connString
sqlquery.Connection = sqlconn
sqlconn.Open()
sqlquery.CommandText = "UPDATE REGISTEROK
SET[Password]='"&pass.Text&"',[Conpassword]='"&cpass.Text&"'WHERE[
Username]='"&user.Text&"';"
DimdrAsOleDbDataReader = cmd.ExecuteReader
Ifdr.Read = FalseThen
MsgBox("User not found!")
ElseIfpass.Text<>cpass.TextThen
MsgBox("Password and Confirm Password does not match!")
Else
sqlquery.ExecuteNonQuery()
MsgBox("Password Successfully Changed!")
Login.Show()
Me.Hide()
sqlconn.Close()
EndIf
```

```
Catch ex AsException
MessageBox.Show(ex.Message)
EndTry
EndSub
PrivateSub Form2_Load(ByVal sender AsSystem.Object, ByVal e
AsSystem.EventArgs) HandlesMyBase.Load
Me.Refresh()
Login.Hide()
EndSub

PrivateSub Button2_Click(ByVal sender AsSystem.Object, ByVal e
AsSystem.EventArgs) Handles Button2.Click
Login.Show()
EndSub
EndClass
```

MDI FORM(HOMEPAGE)-

```
ImportsSystem.Windows.Forms
PublicClassMDIParent1

PrivateSubCToolStripMenuItem_Click(ByVal sender AsSystem.Object,
ByVal e AsSystem.EventArgs) HandlesCToolStripMenuItem.Click
C.MdiParent = Me
C.Show()
EndSub

PrivateSub MDIParent1_Load(ByVal sender AsSystem.Object, ByVal e
AsSystem.EventArgs) HandlesMyBase.Load
Registration.Hide()
Login.Hide()
h1.MdiParent = Me
CPLS.MdiParent = Me
h1.Show()
C.Hide()
CPLS.Hide()
csharp.Hide()
HTML.Hide()
PHP.Hide()
JAVA.Hide()
PYTHON.Hide()EndSub
```

```
PrivateSub CToolStripMenuItem1_Click(ByVal sender AsSystem.Object,
ByVal e AsSystem.EventArgs) Handles CToolStripMenuItem1.Click
CPLS.MdiParent = Me
CPLS.Show()
EndSub

PrivateSub MenuStrip1_ItemClicked(ByVal sender AsSystem.Object, ByVal
e AsSystem.Windows.Forms.ToolStripItemClickedEventArgs) Handles
MenuStrip1.ItemClicked
'Dim newMDIChild As New CPLS()
EndSub


PrivateSub CToolStripMenuItem2_Click(ByVal sender AsSystem.Object,
ByVal e AsSystem.EventArgs) Handles CToolStripMenuItem2.Click
csharp.MdiParent = Me
csharp.Show()
EndSub

PrivateSubVBToolStripMenuItem_Click(ByVal sender AsSystem.Object,
ByVal e AsSystem.EventArgs) HandlesVBToolStripMenuItem.Click
VB.MdiParent = Me
VB.Show()
EndSub

PrivateSubVBNetToolStripMenuItem_Click(ByVal sender AsSystem.Object,
ByVal e AsSystem.EventArgs) HandlesVBNetToolStripMenuItem.Click
VB_NET.MdiParent = Me
VB_NET.Show()
EndSub

PrivateSubJAVAToolStripMenuItem_Click(ByVal sender AsSystem.Object,
ByVal e AsSystem.EventArgs) HandlesHTMLLToolStripMenuItem.Click
HTML.MdiParent = Me
HTML.Show()
EndSub

PrivateSubPHPToolStripMenuItem_Click(ByVal sender AsSystem.Object,
ByVal e AsSystem.EventArgs) HandlesPHPToolStripMenuItem.Click
PHP.MdiParent = Me
PHP.Show()
EndSub
```

```
PrivateSubPYTHONToolStripMenuItem_Click(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
HandlesPYTHONToolStripMenuItem.Click
PYTHON.MdiParent = Me
PYTHON.Show()
EndSub

PrivateSubSQLToolStripMenuItem_Click(ByVal sender AsSystem.Object,
ByVal e AsSystem.EventArgs) HandlesSQLToolStripMenuItem.Click
SQL.MdiParent = Me
SQL.Show()
EndSub

PrivateSubHTMLToolStripMenuItem_Click(ByVal sender AsSystem.Object,
ByVal e AsSystem.EventArgs) HandlesHTMLToolStripMenuItem.Click
JAVA.MdiParent = Me
JAVA.Show()
EndSub

PrivateSub Button1_Click(ByVal sender AsSystem.Object, ByVal e
AsSystem.EventArgs) Handles Button1.Click
C.Hide()
CPLS.Hide()
csharp.Hide()
HTML.Hide()
JAVA.Hide()
PYTHON.Hide()
PHP.Hide()
VB.Hide()
VB_NET.Hide()
SQL.Hide()
FEEDBACK.Hide()
Login.Show()
EndSub

PrivateSubFEEDBACKToolStripMenuItem_Click(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
HandlesFEEDBACKToolStripMenuItem.Click
FEEDBACK.MdiParent = Me
FEEDBACK.Show()
EndSub
EndClass
```

C PROGAMMING FORM-

```
ImportsSystem.Data.OleDb
ImportsSystem.Data
ImportsSystem.Data.DataTable
Imports System.IO
PublicClassC
Private conn AsNewOleDbConnection("
Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=C:\Users\admin\Desktop\Project\database\Maindatabase.mdb;")
DimdrAsOleDbDataReader
Dim ds AsDataSet
Dim da AsOleDbDataAdapter
Dim path AsString

PrivateSubC_Leave(ByVal sender AsObject, ByVal e AsSystem.EventArgs)
HandlesMe.Leave
 RichTextBox1.Hide()
EndSub

PrivateSubC_Load(ByVal sender AsSystem.Object, ByVal e
AsSystem.EventArgs) HandlesMyBase.Load
cmbarray.Text = ""
cmbcontrol.Text = ""
cmbfile.Text = ""
cmbfunction.Text = ""
cmbinput.Text = ""
cmbpointer.Text = ""
cmbstring.Text = ""
cmbstructure.Text = ""
       RichTextBox1.Text = ""
Me.Refresh()
Me.MdiParent = MDIParent1
Me.Dock = DockStyle.Fill
VB.Hide()
PYTHON.Hide()
CPLS.Hide()
HTML.Hide()
SQL.Hide()
PHP.Hide()
JAVA.Hide()
csharp.Hide()
VB_NET.Hide()
 Registration.Hide()
```

```
conn.Open()
'load table of C array
cmbarray.Items.Clear() 'clear item into combobox
DimcmdAsNewOleDbCommand
cmd.CommandText = "select * from CARRAY"
cmd.Connection = conn
dr = cmd.ExecuteReader
Whiledr.Read
cmbarray.Items.Add(dr.GetString(1))  'load data column with name emp
EndWhile
dr.Close()
'load table of C control statement
cmbcontrol.Items.Clear() 'clear item into combobox
Dim cmd1 AsNewOleDbCommand
      cmd1.CommandText = "select * from CCONTROLSTATEMENT"
      cmd1.Connection = conn
dr = cmd1.ExecuteReader
Whiledr.Read
cmbcontrol.Items.Add(dr.GetString(1))  'load data column with name emp
EndWhile
'load table of C files
cmbfile.Items.Clear() 'clear item into combobox
Dim cmd2 AsNewOleDbCommand
      cmd2.CommandText = "select * from CFILES"
      cmd2.Connection = conn
dr = cmd2.ExecuteReader
Whiledr.Read
cmbfile.Items.Add(dr.GetString(1))  'load data column with name emp
EndWhile
'load table of C function
cmbfunction.Items.Clear() 'clear item into combobox
Dim cmd3 AsNewOleDbCommand
      cmd3.CommandText = "select * from CFUNCTIONS"
      cmd3.Connection = conn
dr = cmd3.ExecuteReader
Whiledr.Read
cmbfunction.Items.Add(dr.GetString(1))  'load data column with name emp
EndWhile
'load table of C input an output
cmbinput.Items.Clear() 'clear item into combobox
Dim cmd4 AsNewOleDbCommand
      cmd4.CommandText = "select * from CINPUTOUTPUT"
      cmd4.Connection = conn
```

```
dr = cmd4.ExecuteReader
Whiledr.Read
cmbinput.Items.Add(dr.GetString(1))  'load data column with name emp
EndWhile
'load table to of C pointer
cmbpointer.Items.Clear() 'clear item into combobox
Dim cmd5 AsNewOleDbCommand
        cmd5.CommandText = "select * from CPOINTER"
        cmd5.Connection = conn
dr = cmd5.ExecuteReader
Whiledr.Read
cmbpointer.Items.Add(dr.GetString(1))  'load data column with name emp
EndWhile
'load table of C string
cmbstring.Items.Clear() 'clear item into combobox
Dim cmd6 AsNewOleDbCommand
        cmd6.CommandText = "select * from CSTRING"
        cmd6.Connection = conn
dr = cmd6.ExecuteReader
Whiledr.Read
cmbstring.Items.Add(dr.GetString(1))  'load data column with name emp

EndWhile
'load table of C structure
cmbstructure.Items.Clear() 'clear item into combobox
Dim cmd7 AsNewOleDbCommand
        cmd7.CommandText = "select * from CSTRUCTUREUNION"
        cmd7.Connection = conn
dr = cmd7.ExecuteReader
Whiledr.Read
cmbstructure.Items.Add(dr.GetString(1))  'load data column with name emp
EndWhile
EndSub

PrivateSubcmbarray_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs) Handlescmbarray.KeyPress
e.Handled = True
EndSub
PrivateSub ComboBox1_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbarray.SelectedIndexChanged
RichTextBox1.Show()
conn.Close()
```

```
Dim find AsNewOleDbCommand("Select * from CARRAY where
topic='"&cmbarray.Text&"'", conn)
conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()
Ifch.Read() = TrueThen
        path = ch("path")
        RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
EndIf
conn.Close() EndSub

PrivateSubcmbcontrol_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs)
Handlescmbcontrol.KeyPress
e.Handled = True
EndSub

PrivateSub ComboBox2_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbcontrol.SelectedIndexChanged
    RichTextBox1.Show()
conn.Close()
Dim find AsNewOleDbCommand("Select * from CCONTROLSTATEMENT
where topic='"&cmbcontrol.Text&"'", conn)
conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()
Ifch.Read() = TrueThen
        path = ch("path")
        RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
EndIf
conn.Close()
EndSub

PrivateSubcmbfile_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs) Handlescmbfile.KeyPress
e.Handled = True
EndSub

PrivateSub ComboBox3_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbfile.SelectedIndexChanged
    RichTextBox1.Show()
conn.Close()
```

```
Dim find AsNewOleDbCommand("Select * from CFILES where
topic='"&cmbfile.Text&"'", conn)
conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()
Ifch.Read() = TrueThen
        path = ch("path")
        RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
EndIf
conn.Close()
EndSub

PrivateSubcmbfunction_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs)
Handlescmbfunction.KeyPress
e.Handled = True
EndSub

PrivateSub ComboBox4_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbfunction.SelectedIndexChanged
    RichTextBox1.Show()
conn.Close()
Dim find AsNewOleDbCommand("Select * from CFUNCTIONS where
topic='"&cmbfunction.Text&"'", conn)
conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()
Ifch.Read() = TrueThen
        path = ch("path")
        RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
EndIf
conn.Close()
EndSub
PrivateSubcmbinput_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs) Handlescmbinput.KeyPress
e.Handled = True
EndSub

PrivateSub ComboBox5_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbinput.SelectedIndexChanged
    RichTextBox1.Show()
conn.Close()
```

```
Dim find AsNewOleDbCommand("Select * from CINPUTOUTPUT where
topic='"&cmbinput.Text&"'", conn)
conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()
Ifch.Read() = TrueThen
        path = ch("path")
        RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
EndIf
conn.Close()
EndSub

PrivateSubcmbpointer_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs)
Handlescmbpointer.KeyPress
e.Handled = True
EndSub

PrivateSub ComboBox6_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbpointer.SelectedIndexChanged
    RichTextBox1.Show()
conn.Close()
Dim find AsNewOleDbCommand("Select * from CPOINTER where
topic='"&cmbpointer.Text&"'", conn)
conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()
Ifch.Read() = TrueThen
        path = ch("path")
        RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
EndIf
conn.Close()
EndSub
PrivateSubcmbstring_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs) Handlescmbstring.KeyPress
e.Handled = True
EndSub

PrivateSub ComboBox7_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbstring.SelectedIndexChanged
    RichTextBox1.Show()
conn.Close()
```

```
Dim find AsNewOleDbCommand("Select * from CSTRING where
topic='"&cmbstring.Text&"'", conn)
conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()
Ifch.Read() = TrueThen
        path = ch("path")
         RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
EndIf
conn.Close()
EndSub

PrivateSubcmbstructure_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs)
Handlescmbstructure.KeyPress
e.Handled = True
EndSub

PrivateSub ComboBox8_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbstructure.SelectedIndexChanged
      RichTextBox1.Show()
conn.Close()
Dim find AsNewOleDbCommand("Select * from CSTRUCTUREUNION
where topic='"&cmbstructure.Text&"'", conn)
conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()
Ifch.Read() = TrueThen
         path = ch("path")
         RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
EndIf
conn.Close()
EndSub

PrivateSub ComboBox1_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbarray.Click
cmbfile.Text = ""
cmbcontrol.Text = ""
cmbfunction.Text = ""
cmbinput.Text = ""
cmbpointer.Text = ""
cmbstring.Text = ""
cmbstructure.Text = ""
EndSub
```

```
PrivateSub ComboBox2_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbcontrol.Click
cmbfile.Text = ""
cmbarray.Text = ""
cmbfunction.Text = ""
cmbinput.Text = ""
cmbpointer.Text = ""
cmbstring.Text = ""
cmbstructure.Text = ""
EndSub
PrivateSub ComboBox3_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbfile.Click
cmbarray.Text = ""
cmbcontrol.Text = ""
cmbfunction.Text = ""
cmbinput.Text = ""
cmbpointer.Text = ""
cmbstring.Text = ""
cmbstructure.Text = ""
EndSub
PrivateSub ComboBox4_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbfunction.Click
cmbfile.Text = ""
cmbcontrol.Text = ""
cmbarray.Text = ""
cmbinput.Text = ""
cmbpointer.Text = ""
cmbstring.Text = ""
cmbstructure.Text = ""
EndSub
PrivateSub ComboBox5_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbinput.Click
cmbfile.Text = ""
cmbcontrol.Text = ""
cmbfunction.Text = ""
cmbarray.Text = ""
cmbpointer.Text = ""
cmbstring.Text = ""
cmbstructure.Text = ""
EndSub
PrivateSub ComboBox6_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbpointer.Click
```

```
cmbfile.Text = ""
cmbcontrol.Text = ""
cmbfunction.Text = ""
cmbinput.Text = ""
cmbarray.Text = ""
cmbstring.Text = ""
cmbstructure.Text = ""
EndSub
PrivateSub ComboBox7_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbstring.Click
cmbfile.Text = ""
cmbcontrol.Text = ""
cmbfunction.Text = ""
cmbinput.Text = ""
cmbpointer.Text = ""
cmbarray.Text = ""
cmbstructure.Text = ""
EndSub
PrivateSub ComboBox8_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbstructure.Click
cmbfile.Text = ""
cmbcontrol.Text = ""
cmbfunction.Text = ""
cmbinput.Text = ""
cmbpointer.Text = ""
cmbstring.Text = ""
cmbarray.Text = ""
EndSub
EndClass
```

C++ PROGRAMMING FORM-

```
ImportsSystem.Data.OleDb
ImportsSystem.Data
ImportsSystem.Data.DataTable
Imports System.IO
PublicClassCPLS
Private conn AsNewOleDbConnection("
Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=C:\Users\admin\Desktop\Project\database\Maindatabase.mdb;")
DimdrAsOleDbDataReader
Dim ds AsDataSet
Dim da AsOleDbDataAdapter
Dim path AsString
```

```
PrivateSubCPLS_Leave(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) HandlesMe.Leave
      RichTextBox1.Hide()
EndSub
PrivateSubCPLS_Load(ByVal sender AsSystem.Object, ByVal e
AsSystem.EventArgs) HandlesMyBase.Load
Me.Refresh()
Me.MdiParent = MDIParent1
Me.Dock = DockStyle.Fill
       RichTextBox1.Hide()
Home.Hide()
Registration.Hide()
PYTHON.Hide()
C.Hide()
HTML.Hide()
SQL.Hide()
PHP.Hide()
JAVA.Hide()
csharp.Hide()
VB_NET.Hide()
VB.Hide()
conn.Open()
'load table of CPLUS array
cmbarray.Items.Clear() 'clear item into combobox
DimcmdAsNewOleDbCommand
cmd.CommandText = "select * from CPLUSARRAY"
cmd.Connection = conn
dr = cmd.ExecuteReader
Whiledr.Read
cmbarray.Items.Add(dr.GetString(1))  'load data column with name emp
EndWhile
dr.Close()
'load table of CPLUS cLASSES
cmbclass.Items.Clear() 'clear item into combobox
Dim cmd1 AsNewOleDbCommand
       cmd1.CommandText = "select * from CPLUSCLASS"
       cmd1.Connection = conn
dr = cmd1.ExecuteReader
Whiledr.Read
cmbclass.Items.Add(dr.GetString(1))  'load data column with name emp
EndWhile
'load table of CPLUS CONDITOINAL
```

```
cmbcondition.Items.Clear() 'clear item into combobox
Dim cmd2 AsNewOleDbCommand
      cmd2.CommandText = "select * from
CPLUSCONDITIONALSTATEMENT"
      cmd2.Connection = conn
dr = cmd2.ExecuteReader
Whiledr.Read
cmbcondition.Items.Add(dr.GetString(1))  'load data column with name emp
EndWhile
'load table of CPLUS EXCEPTION
cmbexception.Items.Clear() 'clear item into combobox
Dim cmd3 AsNewOleDbCommand
      cmd3.CommandText = "select * from CPLUSEXCEPTION"
      cmd3.Connection = conn
dr = cmd3.ExecuteReader
Whiledr.Read
cmbexception.Items.Add(dr.GetString(1))  'load data column with name emp
EndWhile
'load table of CPLUS FILES
cmbfiles.Items.Clear() 'clear item into combobox
Dim cmd4 AsNewOleDbCommand
      cmd4.CommandText = "select * from CPLUSFILES"
      cmd4.Connection = conn
dr = cmd4.ExecuteReader
Whiledr.Read
cmbfiles.Items.Add(dr.GetString(1))  'load data column with name emp
EndWhile
'load table to of CPLUS FUNCTION
cmbfunction.Items.Clear() 'clear item into combobox
Dim cmd5 AsNewOleDbCommand
      cmd5.CommandText = "select * from CPLUSFUNCTION"
      cmd5.Connection = conn
dr = cmd5.ExecuteReader
Whiledr.Read
cmbfunction.Items.Add(dr.GetString(1))  'load data column with name emp

EndWhile
'load table of CPLUS INPUT OUTPUT
cmbinput.Items.Clear() 'clear item into combobox
Dim cmd6 AsNewOleDbCommand
      cmd6.CommandText = "select * from CPLUSINPUTOUTPUT"
      cmd6.Connection = conn
```

```
dr = cmd6.ExecuteReader
Whiledr.Read
cmbinput.Items.Add(dr.GetString(1))  'load data column with name emp
EndWhile
'load table of CPLUS REFERENCE
cmbreference.Items.Clear() 'clear item into combobox
Dim cmd7 AsNewOleDbCommand
      cmd7.CommandText = "select * from
CPLUSREFERENCESPOINTERS"
      cmd7.Connection = conn
dr = cmd7.ExecuteReader
Whiledr.Read
cmbreference.Items.Add(dr.GetString(1))  'load data column with name emp
EndWhile
'load table of CPLUS STRING
cmbstring.Items.Clear() 'clear item into combobox
Dim cmd8 AsNewOleDbCommand
      cmd8.CommandText = "select * from CPLUSSTRINGFUNCTION"
      cmd8.Connection = conn
dr = cmd8.ExecuteReader
Whiledr.Read
cmbstring.Items.Add(dr.GetString(1))  'load data column with name emp
EndWhile
EndSub

PrivateSubcmbarray_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs) Handlescmbarray.KeyPress
e.Handled = True
EndSub

PrivateSub ComboBox1_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbarray.SelectedIndexChanged
      RichTextBox1.Show()
conn.Close()
Dim find AsNewOleDbCommand("Select * from CPLUSARRAY where
topic='"&cmbarray.Text&"'", conn)
conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()
Ifch.Read() = TrueThen
         path = ch("path")
         RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
EndIf
```

```
conn.Close()
EndSub

PrivateSubcmbclass_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs) Handlescmbclass.KeyPress
e.Handled = True
EndSub

PrivateSub ComboBox2_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbclass.SelectedIndexChanged
      RichTextBox1.Show()
conn.Close()
Dim find AsNewOleDbCommand("Select * from CPLUSCLASS where
topic='"&cmbclass.Text&"'", conn)
conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()
Ifch.Read() = TrueThen
          path = ch("path")
          RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
EndIf
conn.Close()
EndSub
PrivateSubcmbcondition_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs)
Handlescmbcondition.KeyPress
e.Handled = True
EndSub
PrivateSub ComboBox3_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbcondition.SelectedIndexChanged
      RichTextBox1.Show()
conn.Close()
Dim find AsNewOleDbCommand("Select * from
CPLUSCONDITIONALSTATEMENT where
topic='"&cmbcondition.Text&"'", conn)
conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()
Ifch.Read() = TrueThen
          path = ch("path")
          RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
EndIf
conn.Close()
```

```
EndSub
PrivateSubcmbexception_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs)
Handlescmbexception.KeyPress
e.Handled = True
EndSub
PrivateSub ComboBox4_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbexception.SelectedIndexChanged
      RichTextBox1.Show()
conn.Close()
Dim find AsNewOleDbCommand("Select * from CPLUSEXCEPTION where
topic='"&cmbexception.Text&"'", conn)
conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()
Ifch.Read() = TrueThen
          path = ch("path")
          RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
EndIf
conn.Close()
EndSub
PrivateSubcmbfiles_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs) Handlescmbfiles.KeyPress
e.Handled = True
EndSub

PrivateSub ComboBox5_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbfiles.SelectedIndexChanged
      RichTextBox1.Show()
conn.Close()
Dim find AsNewOleDbCommand("Select * from CPLUSFILES where
topic='"&cmbfiles.Text&"'", conn)
conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()
Ifch.Read() = TrueThen
          path = ch("path")
          RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
EndIf
conn.Close()
EndSub
```

```
PrivateSubcmbfunction_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs)
Handlescmbfunction.KeyPress
e.Handled = True
EndSub

PrivateSub ComboBox6_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbfunction.SelectedIndexChanged
      RichTextBox1.Show()
conn.Close()
Dim find AsNewOleDbCommand("Select * from CPLUSFUNCTION where
topic='"&cmbfunction.Text&"'", conn)
conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()
Ifch.Read() = TrueThen
          path = ch("path")
          RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
EndIf
conn.Close()
EndSub

PrivateSubcmbinput_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs) Handlescmbinput.KeyPress
e.Handled = True
EndSub

PrivateSub ComboBox7_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbinput.SelectedIndexChanged
      RichTextBox1.Show()
conn.Close()
Dim find AsNewOleDbCommand("Select * from CPLUSINPUTOUTPUT
where topic='"&cmbinput.Text&"'", conn)
conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()
Ifch.Read() = TrueThen
          path = ch("path")
          RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
EndIf
conn.Close()
EndSub
```

```
PrivateSubcmbreference_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs)
Handlescmbreference.KeyPress
e.Handled = True
EndSub
PrivateSub ComboBox8_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbreference.SelectedIndexChanged
      RichTextBox1.Show()
conn.Close()
Dim find AsNewOleDbCommand("Select * from
CPLUSREFERENCESPOINTERS where topic='"&cmbreference.Text&"'",
conn)
conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()
Ifch.Read() = TrueThen
         path = ch("path")
         RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
EndIf
conn.Close()
EndSub

PrivateSubcmbstring_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs) Handlescmbstring.KeyPress
e.Handled = True
EndSub

PrivateSub ComboBox9_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbstring.SelectedIndexChanged
      RichTextBox1.Show()
conn.Close()
Dim find AsNewOleDbCommand("Select * from
CPLUSSTRINGFUNCTION where topic='"&cmbstring.Text&"'", conn)
conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()
Ifch.Read() = TrueThen
         path = ch("path")
         RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
EndIf
conn.Close()
EndSub
```

```
PrivateSub ComboBox1_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbarray.Click
cmbcondition.Text = ""
cmbclass.Text = ""
cmbexception.Text = ""
cmbfiles.Text = ""
cmbfunction.Text = ""
cmbinput.Text = ""
cmbreference.Text = ""
cmbstring.Text = ""
EndSub

PrivateSub ComboBox2_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbclass.Click
cmbcondition.Text = ""
cmbarray.Text = ""
cmbexception.Text = ""
cmbfiles.Text = ""
cmbfunction.Text = ""
cmbinput.Text = ""
cmbreference.Text = ""
cmbstring.Text = ""
EndSub
PrivateSub ComboBox3_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbcondition.Click
cmbarray.Text = ""
cmbclass.Text = ""
cmbexception.Text = ""
cmbfiles.Text = ""
cmbfunction.Text = ""
cmbinput.Text = ""
cmbreference.Text = ""
cmbstring.Text = ""
EndSub
PrivateSub ComboBox4_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbexception.Click
cmbcondition.Text = ""
cmbclass.Text = ""
cmbarray.Text = ""
cmbfiles.Text = ""
cmbfunction.Text = ""
cmbinput.Text = ""
```

```
cmbreference.Text = ""
cmbstring.Text = ""
EndSub
PrivateSub ComboBox5_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbfiles.Click
cmbcondition.Text = ""
cmbclass.Text = ""
cmbexception.Text = ""
cmbarray.Text = ""
cmbfunction.Text = ""
cmbinput.Text = ""
cmbreference.Text = ""
cmbstring.Text = ""
EndSub
PrivateSub ComboBox6_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbfunction.Click
cmbcondition.Text = ""
cmbclass.Text = ""
cmbexception.Text = ""
cmbfiles.Text = ""
cmbarray.Text = ""
cmbinput.Text = ""
cmbreference.Text = ""
cmbstring.Text = ""
EndSub
PrivateSub ComboBox7_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbinput.Click
cmbcondition.Text = ""
cmbclass.Text = ""
cmbexception.Text = ""
cmbfiles.Text = ""
cmbfunction.Text = ""
cmbarray.Text = ""
cmbreference.Text = ""
cmbstring.Text = ""
EndSub
PrivateSub ComboBox8_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbreference.Click
cmbcondition.Text = ""
cmbclass.Text = ""
cmbexception.Text = ""
cmbfiles.Text = ""
```

```
cmbfunction.Text = ""
cmbinput.Text = ""
cmbarray.Text = ""
cmbstring.Text = ""
EndSub
PrivateSub ComboBox9_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbstring.Click
cmbcondition.Text = ""
cmbclass.Text = ""
cmbexception.Text = ""
cmbfiles.Text = ""
cmbfunction.Text = ""
cmbinput.Text = ""
cmbarray.Text = ""
cmbreference.Text = ""
EndSub
EndClass
```

HTML FORM-

```
ImportsSystem.Data.OleDb
ImportsSystem.Data
ImportsSystem.Data.DataTable
Imports System.IO

PublicClassHTML
Private conn AsNewOleDbConnection("
Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=C:\Users\admin\Desktop\Project\database\Maindatabase.mdb;")
DimdrAsOleDbDataReader
Dim ds AsDataSet
Dim da AsOleDbDataAdapter
Dim path AsString
Dim image AsString

PrivateSubHTML_Leave(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) HandlesMe.Leave
    RichTextBox1.Hide()
    PictureBox1.Hide()
EndSub
PrivateSubHTML_Load(ByVal sender AsSystem.Object, ByVal e
AsSystem.EventArgs) HandlesMyBase.Load
Me.Refresh()
```

```
Me.MdiParent = MDIParent1
Me.Dock = DockStyle.Fill
Home.Hide()
Registration.Hide()
Login.Hide()
C.Hide()
VB.Hide()
CPLS.Hide()
PYTHON.Hide()
SQL.Hide()
PHP.Hide()
JAVA.Hide()
csharp.Hide()
VB_NET.Hide()
        RichTextBox1.Hide()
        PictureBox1.Hide()
conn.Open()
'load table of html image
cmbimage.Items.Clear() 'clear item into combobox
DimcmdAsNewOleDbCommand
cmd.CommandText = "select * from HTMLIMAGE"
cmd.Connection = conn
dr = cmd.ExecuteReader
Whiledr.Read
cmbimage.Items.Add(dr.GetString(1))  'load data column with name emp
EndWhile
dr.Close()
'load table of html video
cmbvideo.Items.Clear() 'clear item into combobox
Dim cmd1 AsNewOleDbCommand
        cmd1.CommandText = "select * from HTMLVIDEO"
        cmd1.Connection = conn
dr = cmd1.ExecuteReader
Whiledr.Read
cmbvideo.Items.Add(dr.GetString(1))  'load data column with name emp
EndWhile
'load table of html table
cmbtable.Items.Clear() 'clear item into combobox
Dim cmd2 AsNewOleDbCommand
        cmd2.CommandText = "select * from HTMLTABLE"
        cmd2.Connection = conn
dr = cmd2.ExecuteReader
```

```
Whiledr.Read
cmbtable.Items.Add(dr.GetString(1))  'load data column with name emp
EndWhile
'load table of html audio
cmbaudio.Items.Clear() 'clear item into combobox
Dim cmd3 AsNewOleDbCommand
      cmd3.CommandText = "select * from HTMLAUDIO"
      cmd3.Connection = conn
dr = cmd3.ExecuteReader
Whiledr.Read
cmbaudio.Items.Add(dr.GetString(1))  'load data column with name emp
EndWhile
'load table of  html formatting
cmbformatting.Items.Clear() 'clear item into combobox
Dim cmd4 AsNewOleDbCommand
      cmd4.CommandText = "select * from HTMLFORMATTING"
      cmd4.Connection = conn
dr = cmd4.ExecuteReader
Whiledr.Read
cmbformatting.Items.Add(dr.GetString(1))  'load data column with name emp
EndWhile
'load table to html form
cmbform.Items.Clear() 'clear item into combobox
Dim cmd5 AsNewOleDbCommand
      cmd5.CommandText = "select * from HTMLFORMLINKING"
      cmd5.Connection = conn
dr = cmd5.ExecuteReader
Whiledr.Read
cmbform.Items.Add(dr.GetString(1))  'load data column with name emp
EndWhile
EndSub

PrivateSubcmbimage_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs) Handlescmbimage.KeyPress
e.Handled = True
EndSub
PrivateSub ComboBox1_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbimage.SelectedIndexChanged
      RichTextBox1.Show()
      PictureBox1.Show()
conn.Close()
```

```
Dim find AsNewOleDbCommand("Select * from HTMLIMAGE where
topic='"&cmbimage.Text&"'", conn)
conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()

Ifch.Read() = TrueThen
        path = ch("path")
        image = ch("IMAGE")
        RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
        PictureBox1.Image = System.Drawing.Image.FromFile(image)
EndIf
conn.Close()
EndSub

PrivateSubcmbvideo_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs) Handlescmbvideo.KeyPress
e.Handled = True
EndSub

PrivateSub ComboBox2_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbvideo.SelectedIndexChanged
RichTextBox1.Show()
      PictureBox1.Show()
conn.Close()
Dim find AsNewOleDbCommand("Select * from HTMLVIDEO where
topic='"&cmbvideo.Text&"'", conn)

conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()

Ifch.Read() = TrueThen
        path = ch("path")
        image = ch("IMAGE")
        RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
        PictureBox1.Image = System.Drawing.Image.FromFile(image)
EndIf
conn.Close()
EndSub

PrivateSubcmbtable_KeyPress(ByVal sender AsObject, ByVal e
```

```
AsSystem.Windows.Forms.KeyPressEventArgs) Handlescmbtable.KeyPress
e.Handled = True
EndSub

PrivateSub ComboBox3_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbtable.SelectedIndexChanged
      RichTextBox1.Show()
      PictureBox1.Show()
conn.Close()
Dim find AsNewOleDbCommand("Select * from HTMLTABLE where
topic='"&cmbtable.Text&"'", conn)

conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()
Ifch.Read() = TrueThen
         path = ch("path")
         image = ch("IMAGE")
         RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
         PictureBox1.Image = System.Drawing.Image.FromFile(image)
EndIf
conn.Close()
EndSub

PrivateSubcmbform_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs) Handlescmbform.KeyPress
e.Handled = True
EndSub
PrivateSubcmbform_SelectedIndexChanged(ByVal sender AsSystem.Object,
ByVal e AsSystem.EventArgs) Handlescmbform.SelectedIndexChanged
      RichTextBox1.Show()
      PictureBox1.Show()
conn.Close()
Dim find AsNewOleDbCommand("Select * from HTMLFORMLINKING
where topic='"&cmbform.Text&"'", conn)

conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()

Ifch.Read() = TrueThen
         path = ch("path")
         image = ch("IMAGE")
```

```
 RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
        PictureBox1.Image = System.Drawing.Image.FromFile(image)
EndIf
conn.Close()
EndSub

PrivateSubcmbaudio_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs) Handlescmbaudio.KeyPress
e.Handled = True
EndSub
PrivateSub ComboBox4_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbaudio.SelectedIndexChanged
      RichTextBox1.Show()
      PictureBox1.Show()
conn.Close()
Dim find AsNewOleDbCommand("Select * from HTMLAUDIO where
topic='"&cmbaudio.Text&"'", conn)

conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()
Ifch.Read() = TrueThen
        path = ch("path")
        image = ch("IMAGE")
RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
        PictureBox1.Image = System.Drawing.Image.FromFile(image)
EndIf
conn.Close()
EndSub

PrivateSubcmbformatting_KeyPress(ByVal sender AsObject, ByVal e
AsSystem.Windows.Forms.KeyPressEventArgs)
Handlescmbformatting.KeyPress
e.Handled = True
EndSub

PrivateSubcmbformatting_SelectedIndexChanged(ByVal sender
AsSystem.Object, ByVal e AsSystem.EventArgs)
Handlescmbformatting.SelectedIndexChanged
      RichTextBox1.Show()
      PictureBox1.Show()
conn.Close()
```

```
Dim find AsNewOleDbCommand("Select * from HTMLFORMATTING

where topic='"&cmbformatting.Text&"'", conn)

conn.Open()
DimchAsOleDbDataReader = find.ExecuteReader()

Ifch.Read() = TrueThen
        path = ch("path")
        image = ch("IMAGE")
        RichTextBox1.Text = My.Computer.FileSystem.ReadAllText(path)
        PictureBox1.Image = System.Drawing.Image.FromFile(image)
EndIf

conn.Close()
EndSub

PrivateSub ComboBox1_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbimage.Click
cmbtable.Text = ""
cmbvideo.Text = ""
cmbaudio.Text = ""
cmbformatting.Text = ""
cmbform.Text = ""
EndSub
PrivateSub ComboBox2_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbvideo.Click
cmbtable.Text = ""
cmbaudio.Text = ""
cmbformatting.Text = ""
cmbform.Text = ""
cmbimage.Text = ""
EndSub

PrivateSub ComboBox3_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbtable.Click
cmbvideo.Text = ""
cmbaudio.Text = ""
cmbformatting.Text = ""
cmbform.Text = ""
cmbimage.Text = ""
EndSub
```

```
PrivateSub ComboBox4_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbaudio.Click
cmbtable.Text = ""
cmbvideo.Text = ""
cmbformatting.Text = ""
cmbform.Text = ""

cmbimage.Text = ""

EndSub
PrivateSub ComboBox5_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbformatting.Click
cmbtable.Text = ""
cmbvideo.Text = ""
cmbaudio.Text = ""

cmbform.Text = ""

cmbimage.Text = ""

EndSub
PrivateSub ComboBox6_Click(ByVal sender AsObject, ByVal e
AsSystem.EventArgs) Handlescmbform.Click
cmbtable.Text = ""
cmbvideo.Text = ""
cmbaudio.Text = ""
cmbformatting.Text = ""
cmbimage.Text = ""
EndSub
EndClass
```

FEEDBACK FORM-

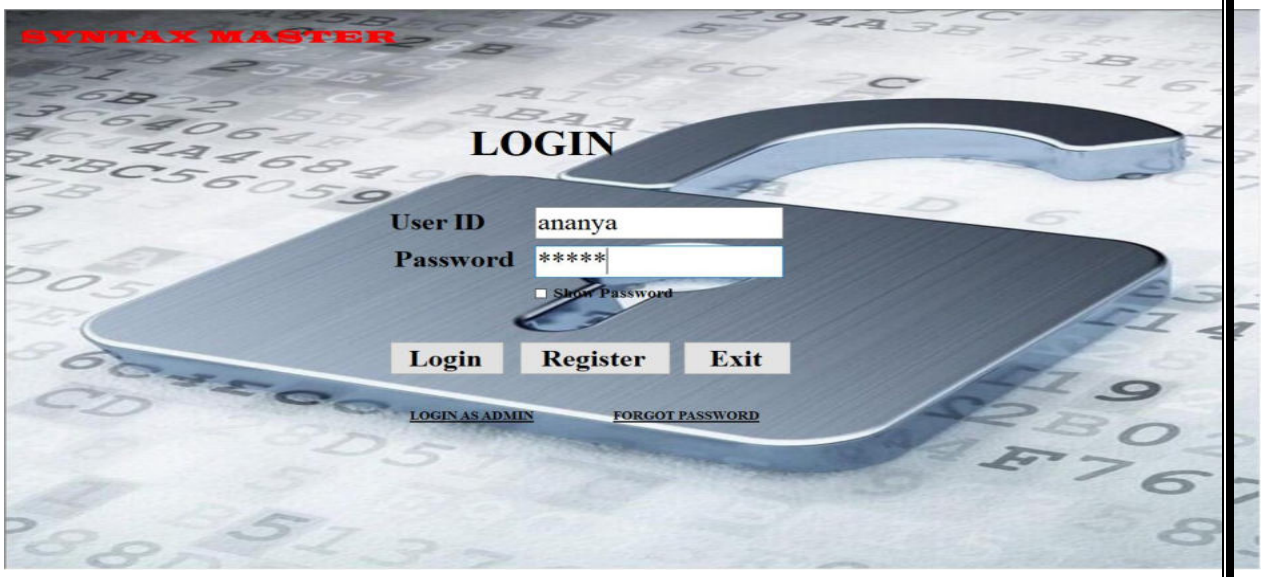
```
ImportsSystem.Text.RegularExpressions
ImportsSystem.Data.OleDb
PublicClassFEEDBACK
Dim conn AsNewOleDbConnection
DimdbpAsString = " Provider=Microsoft.ACE.OLEDB.12.0;"
DimdbsrcAsString = "Data
Source=C:\Users\admin\Desktop\Project\database\Maindatabase.mdb;"
DimadapAsOleDbDataAdapter
Dim ds AsDataSet
```

```
PrivateSub Form1_Load(ByVal sender AsSystem.Object, ByVal e
AsSystem.EventArgs) HandlesMyBase.Load
conn.ConnectionString = dbp&dbsrc
Me.Refresh()
Me.MdiParent = MDIParent1
Me.Dock = DockStyle.Fill
JAVA.Hide()
Home.Hide()
Login.Hide()
Registration.Hide()
PYTHON.Hide()
C.Hide()
HTML.Hide()
SQL.Hide()
PHP.Hide()
csharp.Hide()
CPLS.Hide()
VB_NET.Hide()
EndSub

PrivateSub Button1_Click(ByVal sender AsSystem.Object, ByVal e
AsSystem.EventArgs) Handles Button1.Click
    ds = NewDataSet
adap = NewOleDbDataAdapter("Insert into
[feedback]([NAME],[EMAIL],[content]) values
"&"('"&txtname.Text&"','"&txtmail.Text&"' ,'"&txtmess.Text&"' )", conn)
adap.Fill(ds, "feedback")
txtname.Text = ""
txtmail.Text = ""
txtmess.Text = ""
MsgBox("THANK YOU FOR YOUR FEEDBACK!")
    Button1.Focus()

EndSub
EndClass
```

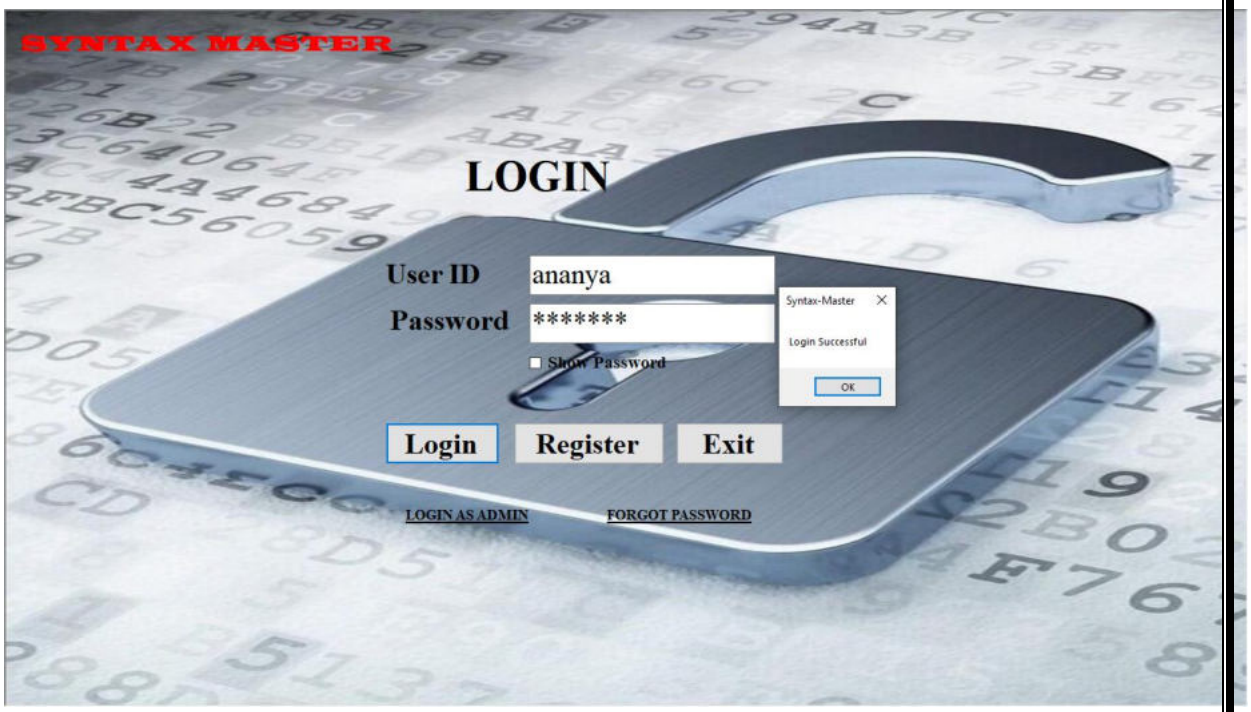
INPUT AND OUTPUT SCREEN

LOGIN FORM

INPUT-

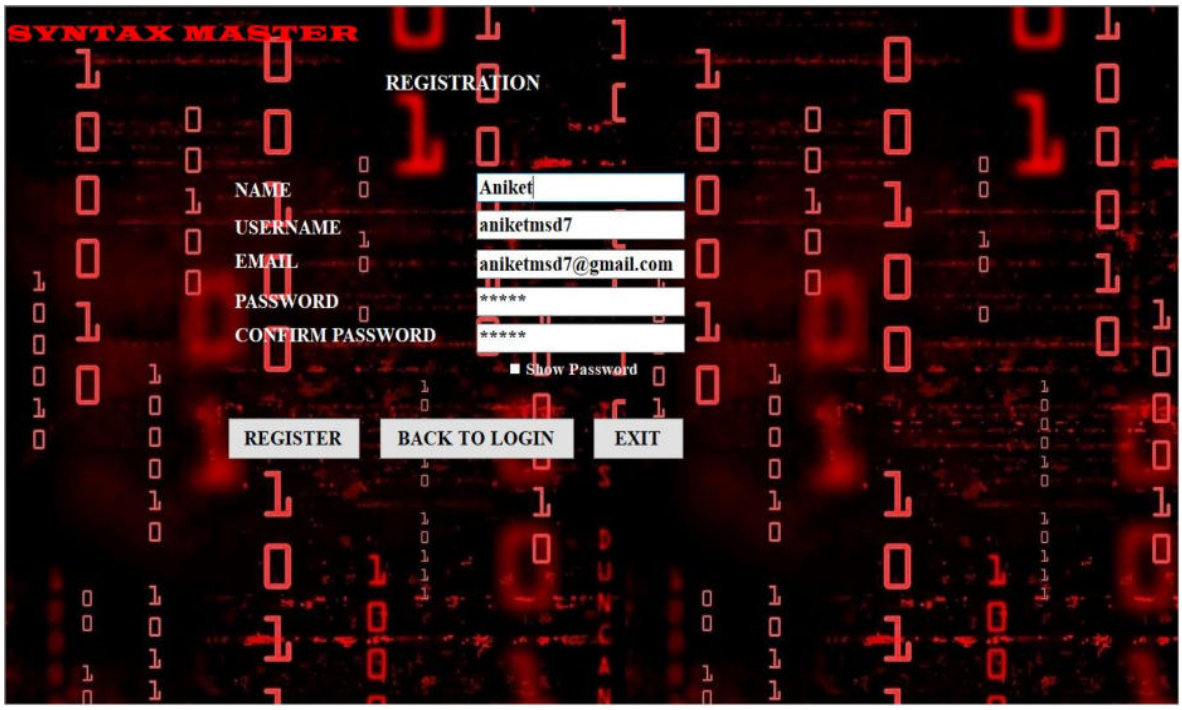


OUTPUT-

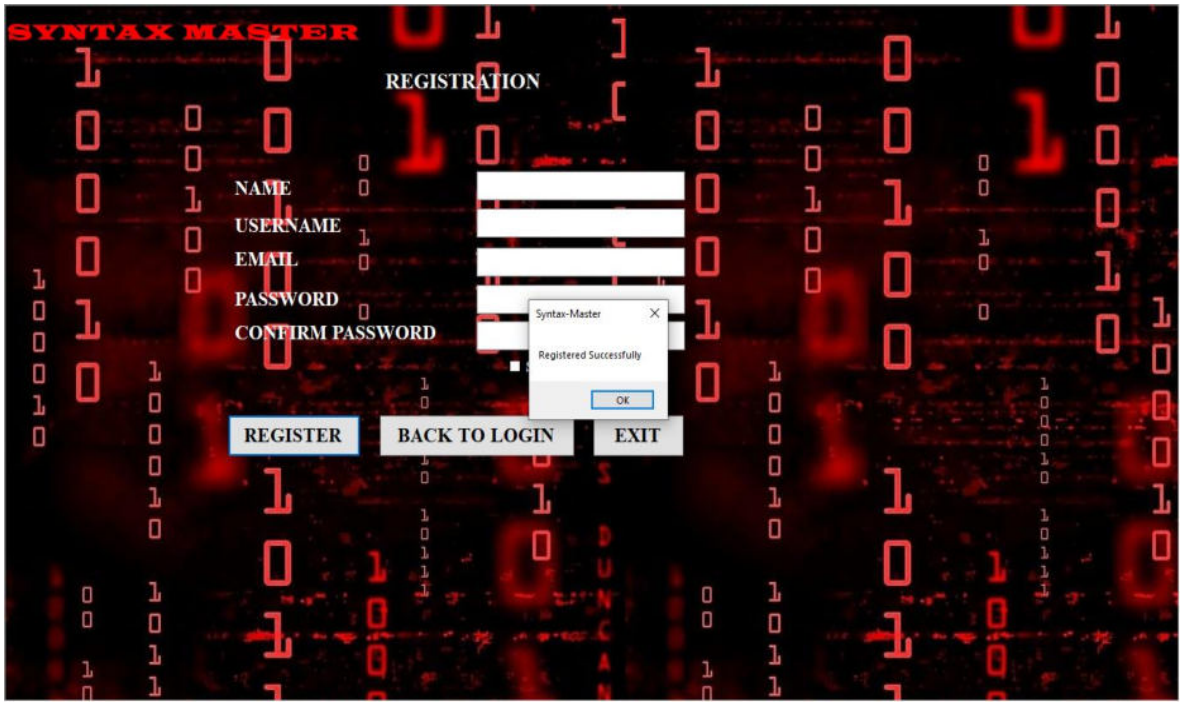


REGISTRATION FORM-

INPUT-



OUTPUT-



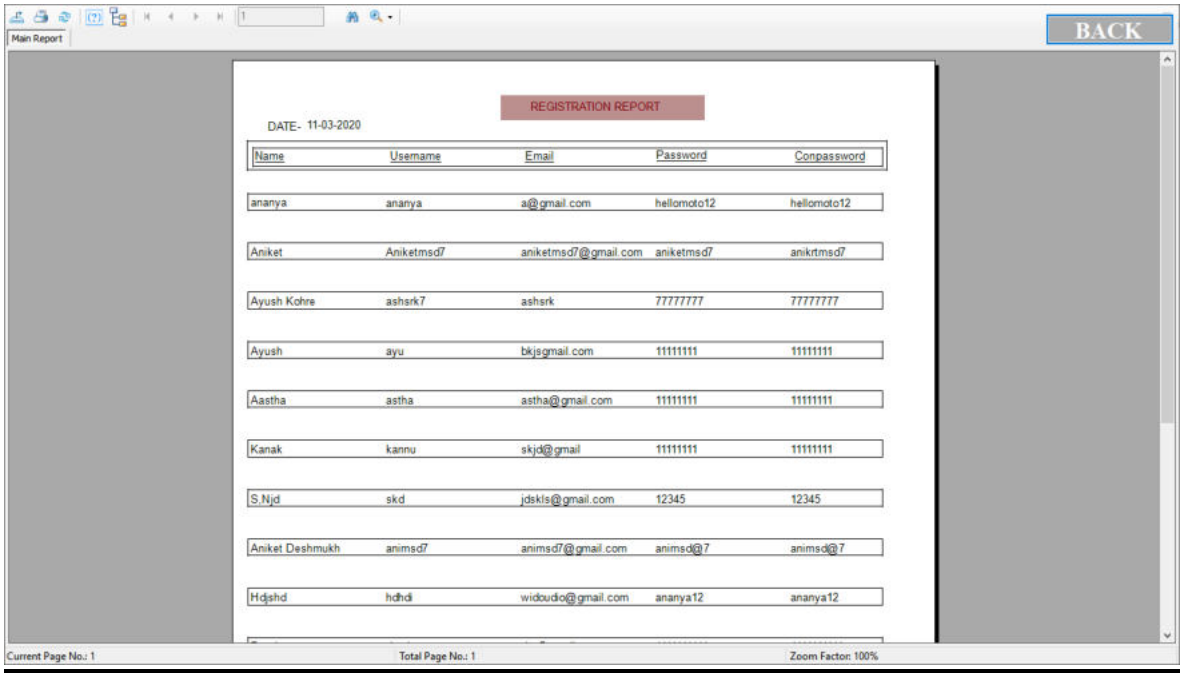
ADMIN -

INPUT-

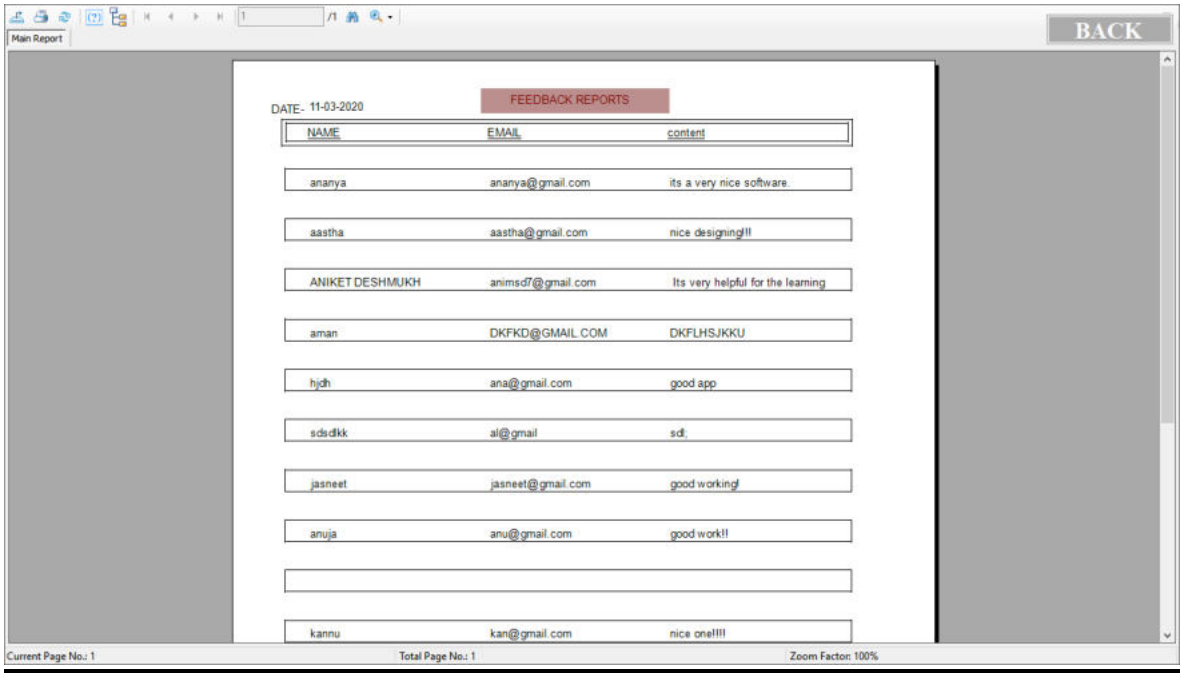
OUTPUT-

REPORTS-

REGISTRATION REPORTS-



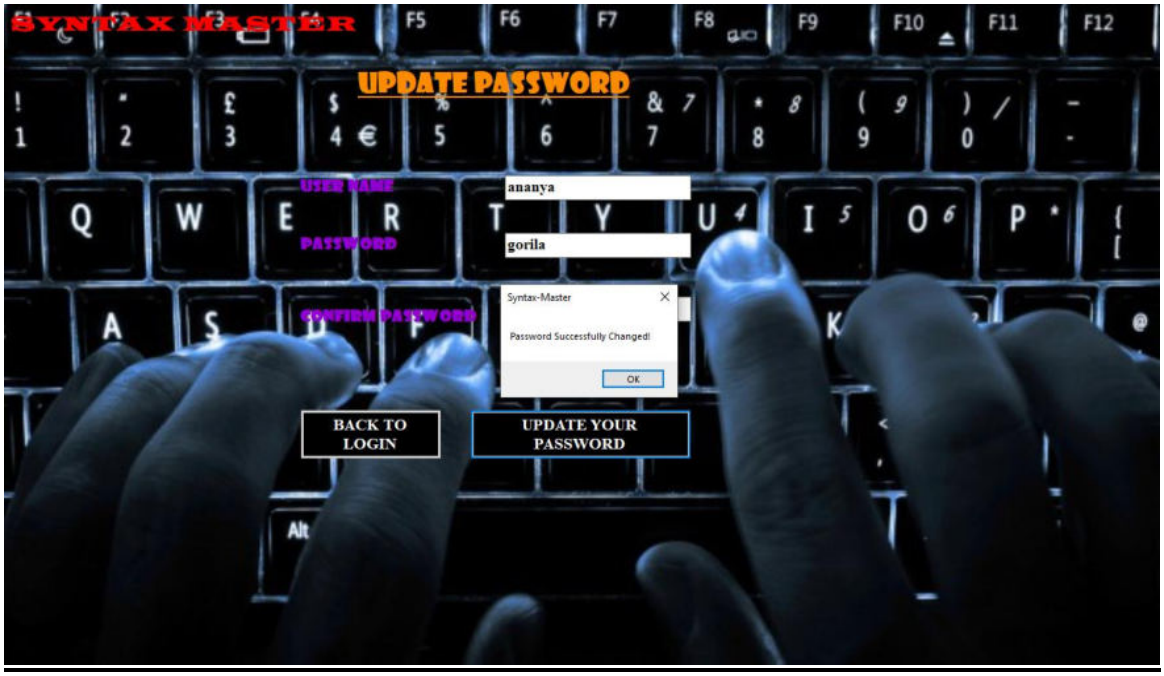
FEEDBACK REPORTS-



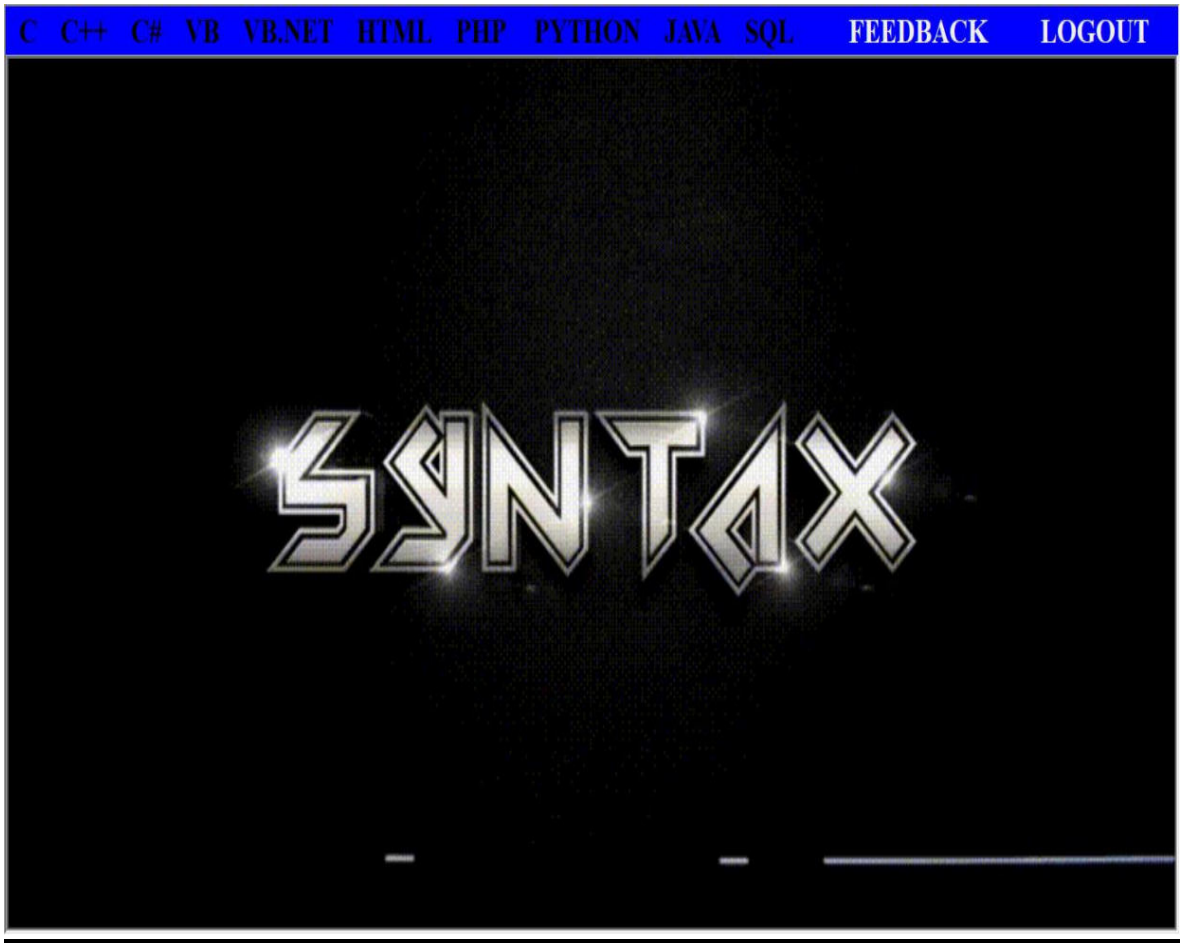
UPDATE PASSWORD-

INPUT-

OUTPUT-

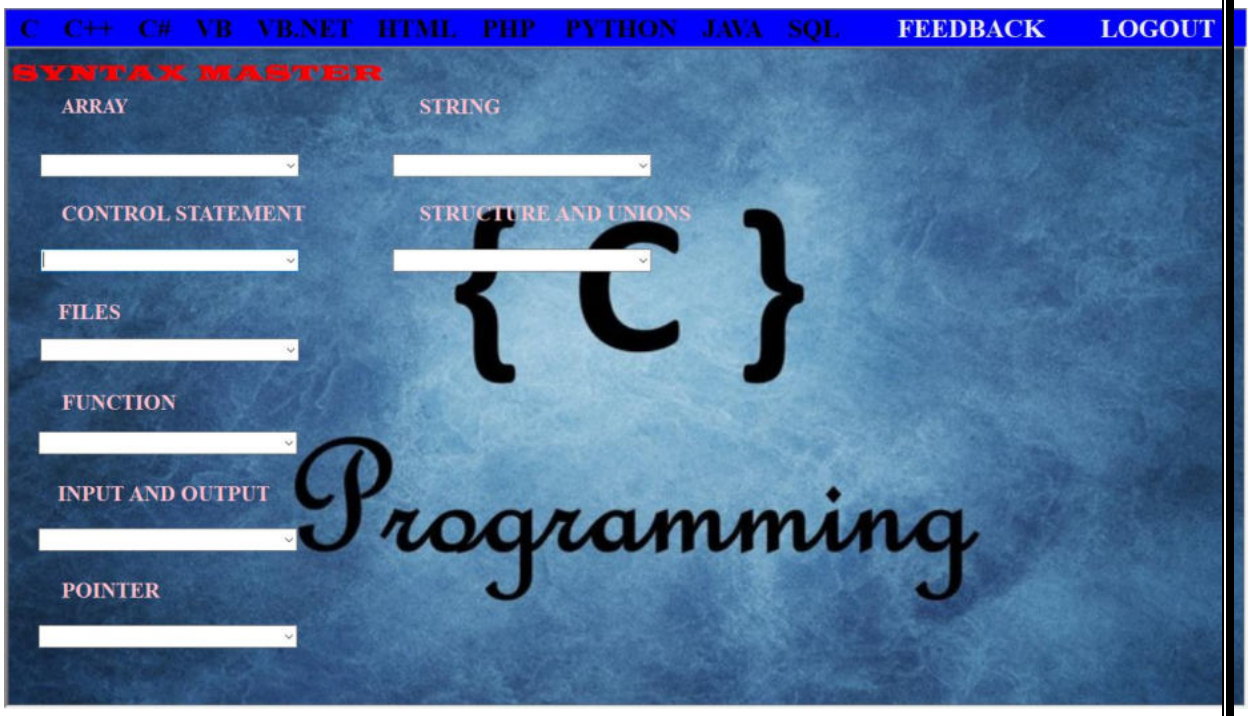


HOMEPAGE-

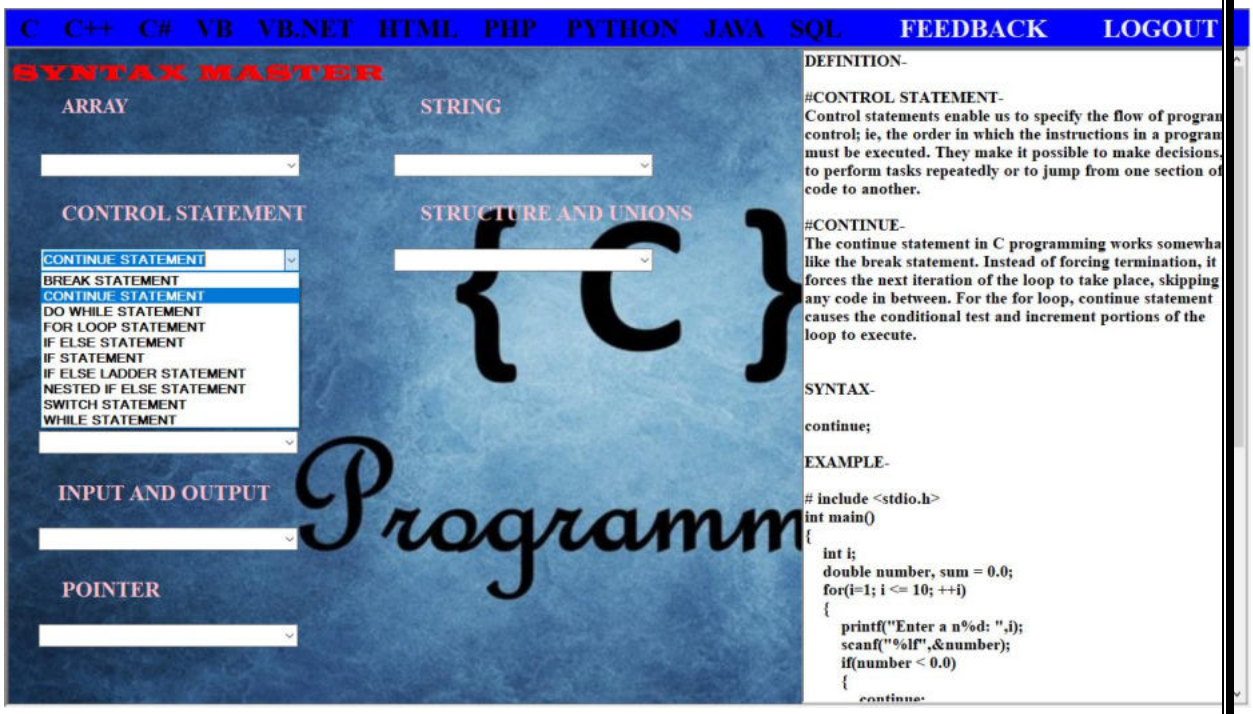

C C++ C# VB VB.NET HTML PHP PYTHON JAVA SQL FEEDBACK LOGOUT
SYNTAX

C PROGRAMMING FORM-

INPUT-



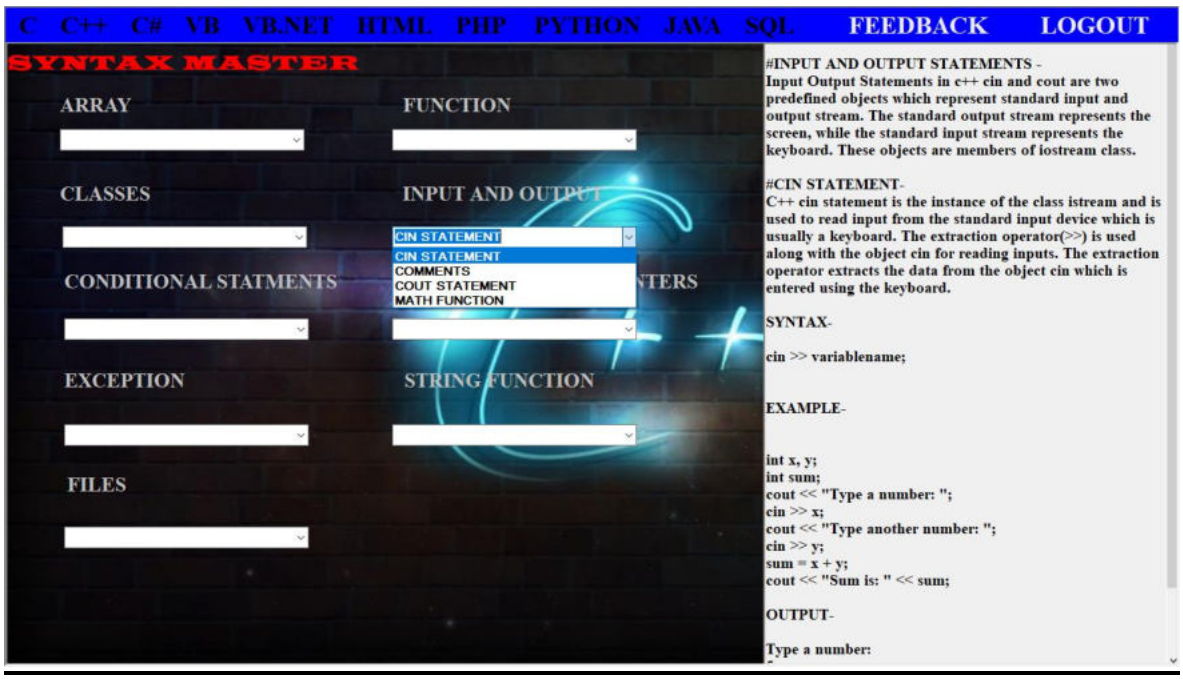
OUTPUT-



C++ PROGRAMMING FORM-

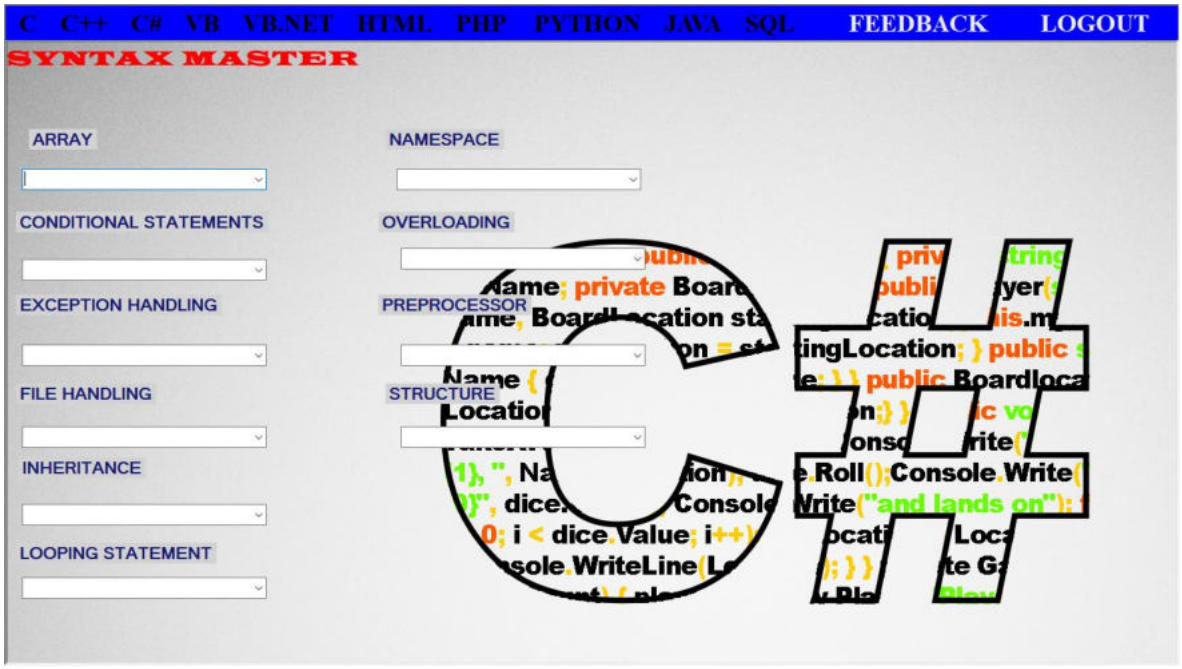
INPUT-

OUTPUT-

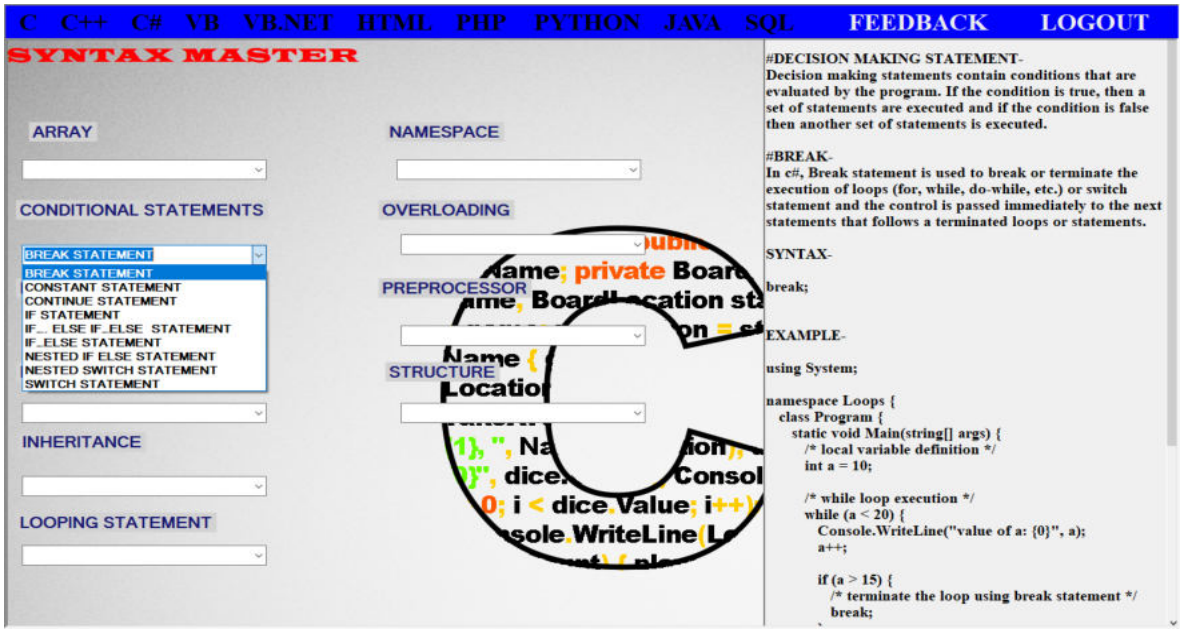


C# PROGRAMMING FORM-

INPUT-

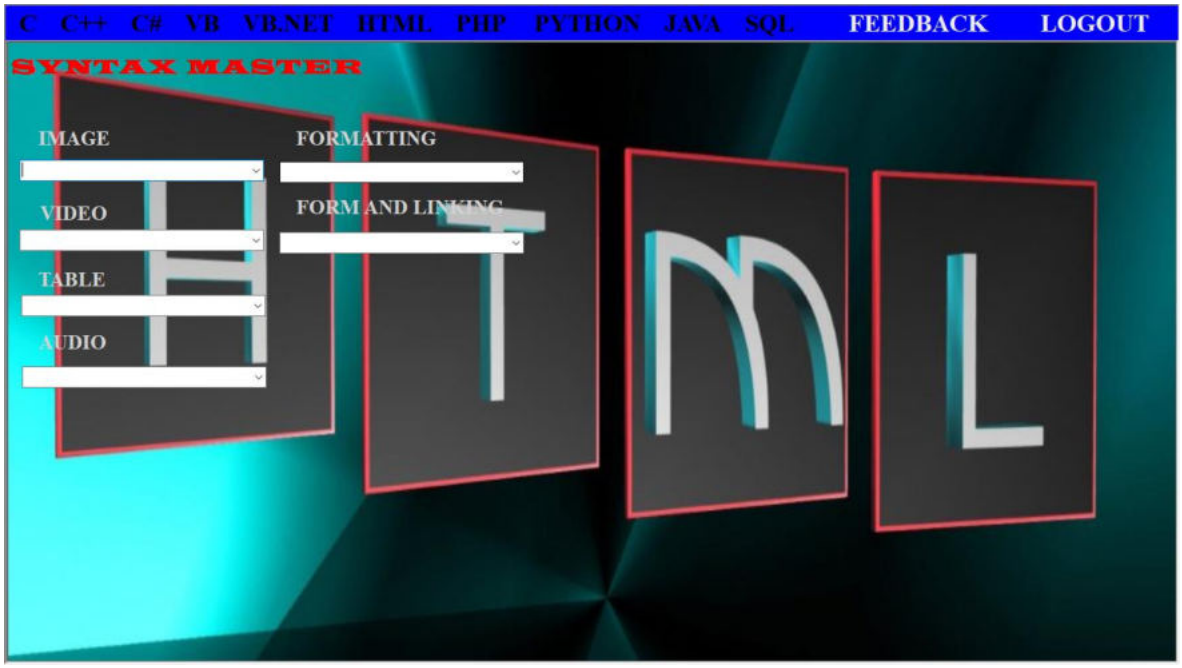


OUTPUT-

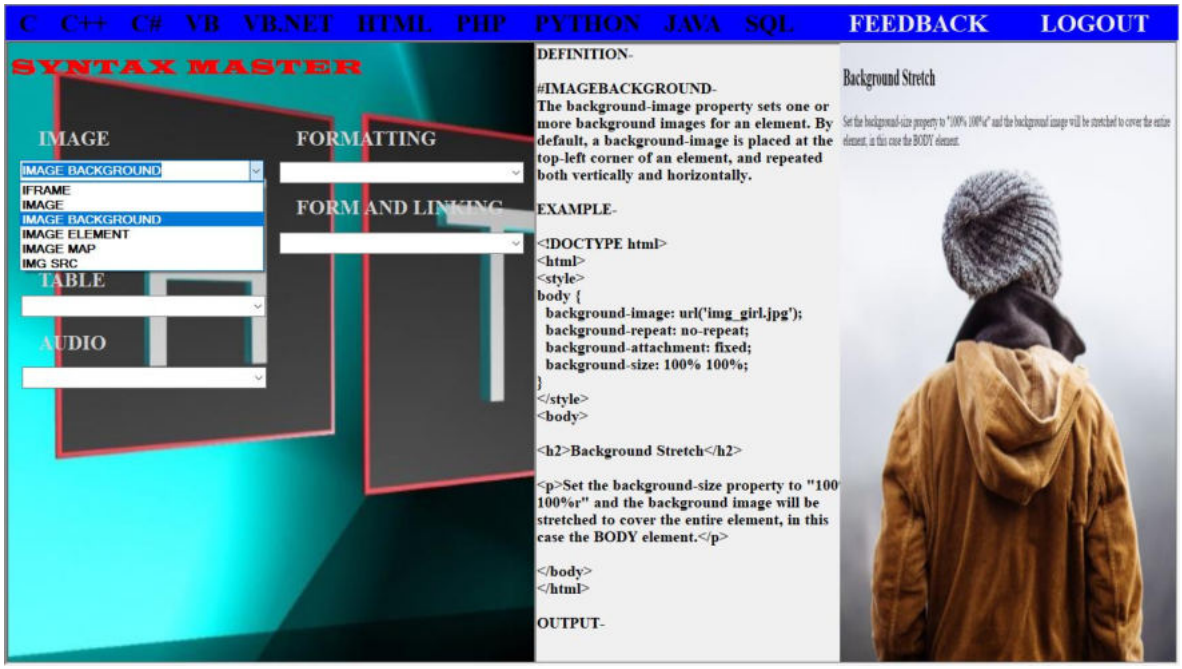


HTML FORM-

INPUT-



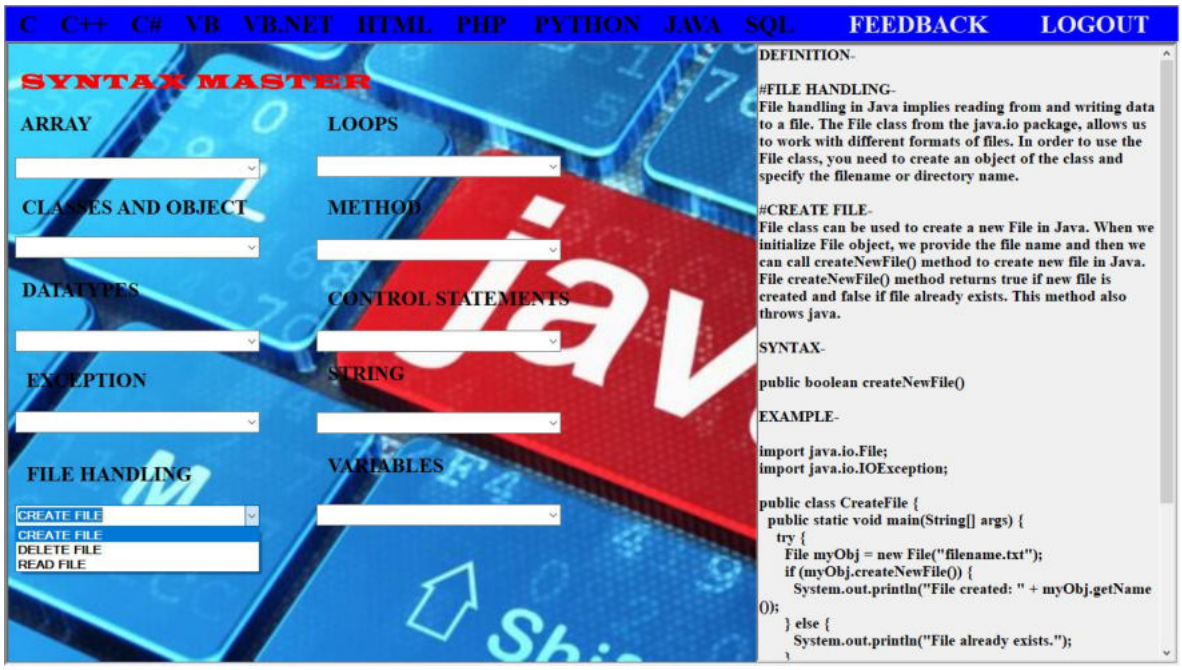
OUTPUT-



JAVA FORM-

INPUT-

OUTPUT-



PHP FORM-

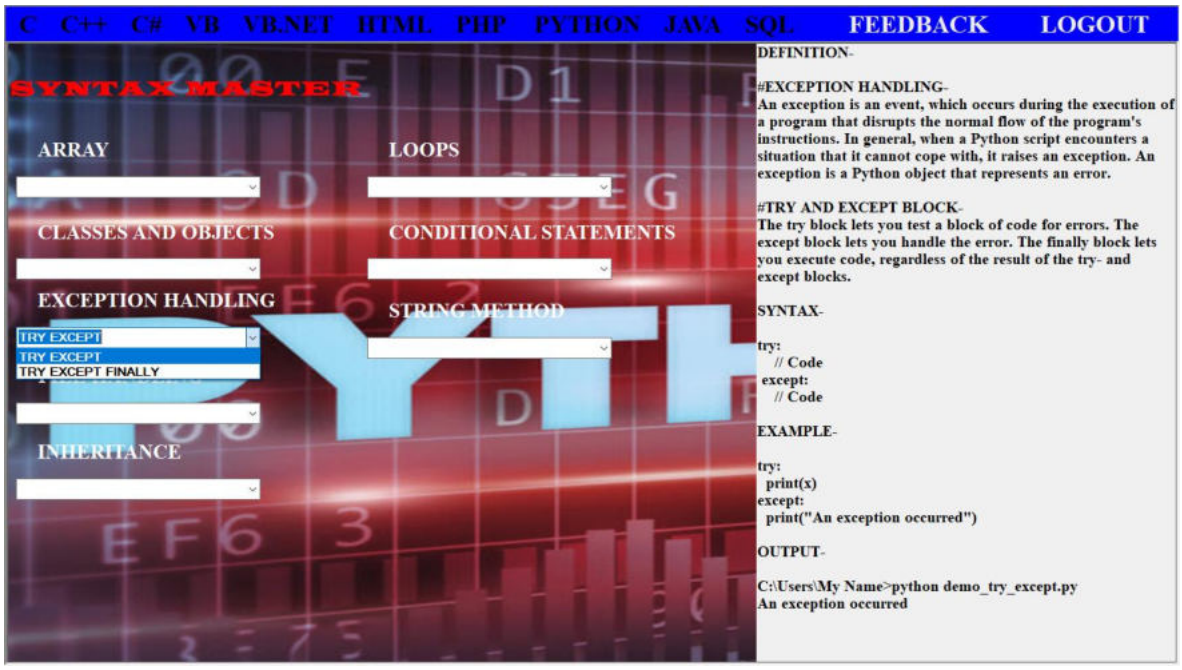
INPUT-

OUTPUT-

PYTHON FORM-

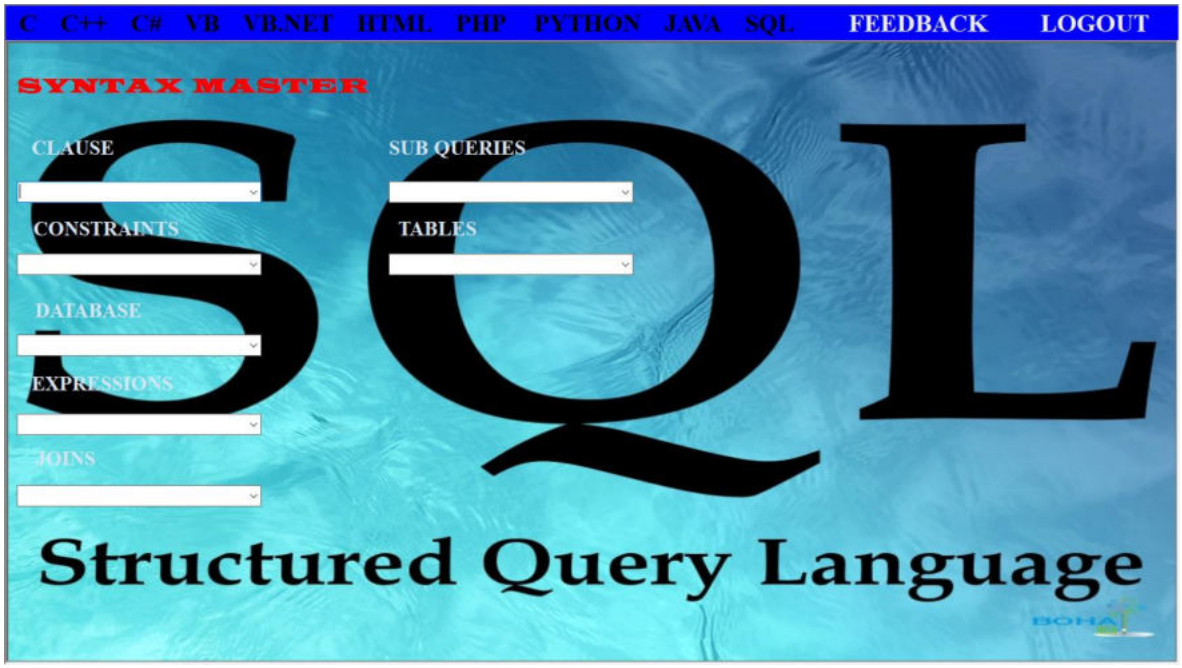
INPUT-

OUTPUT-

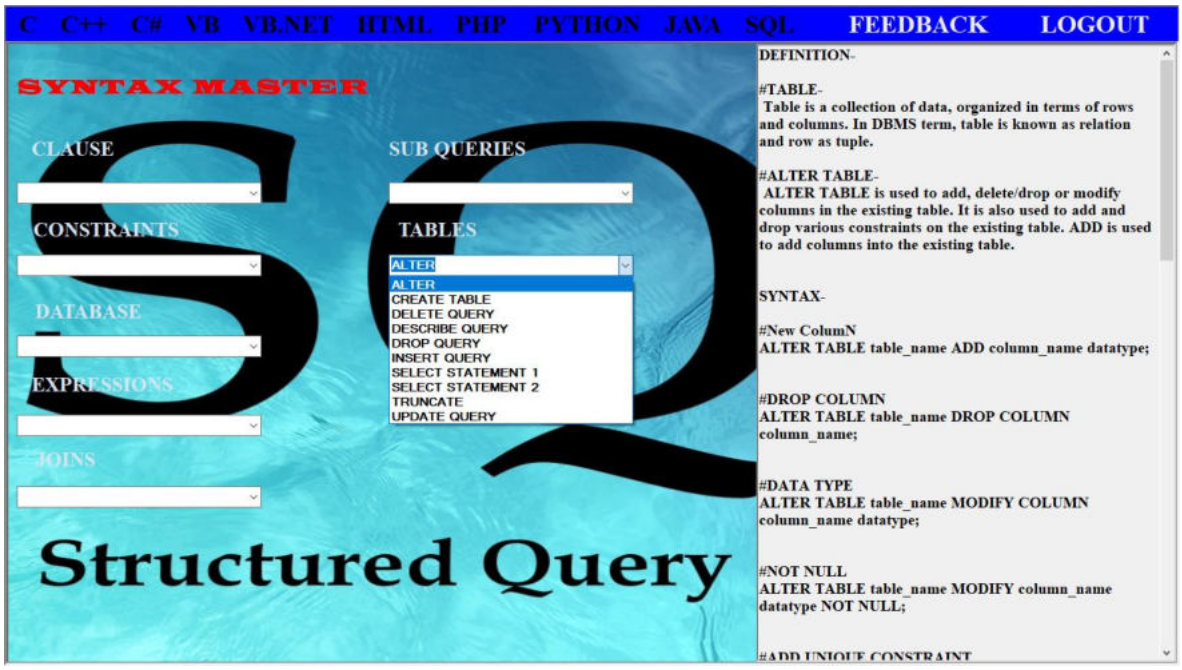


SQL FORM-

INPUT-



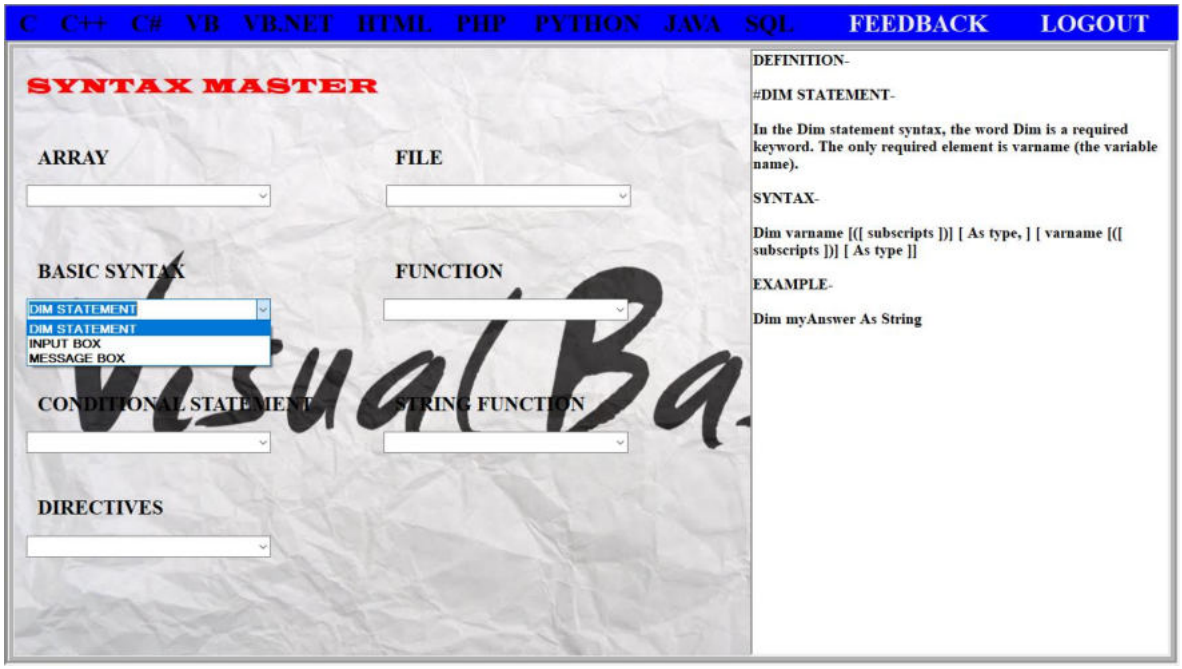
OUTPUT-



VB FORM-

INPUT-

OUTPUT-



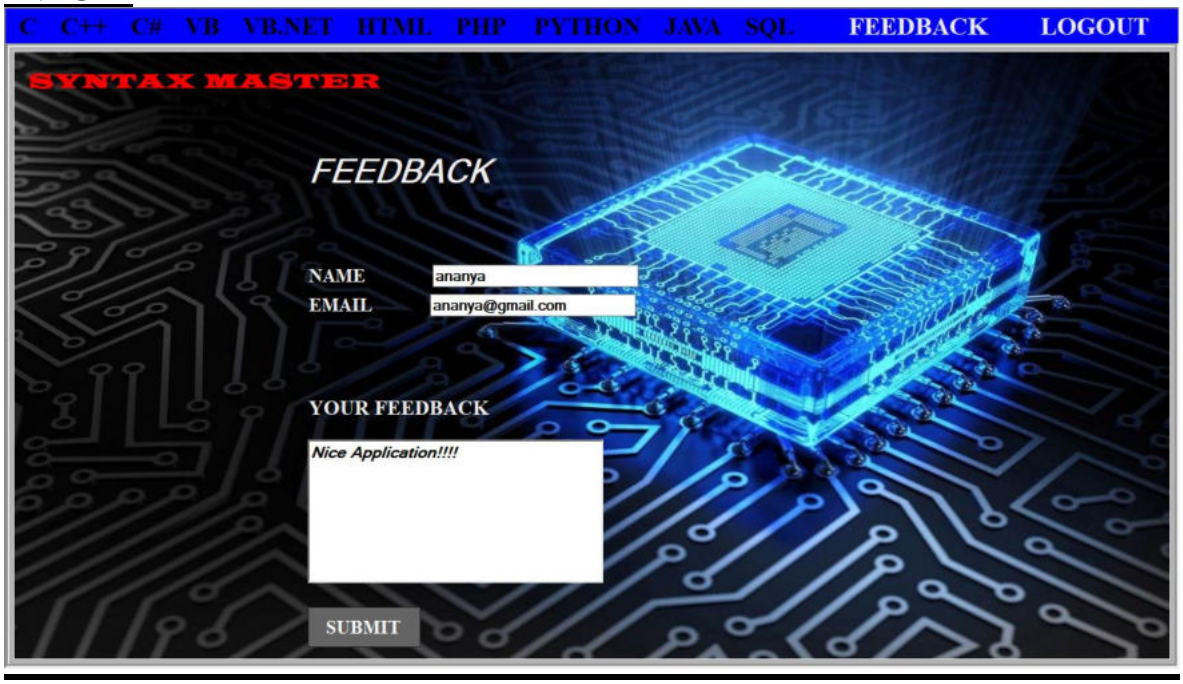
VB.NET FORM-

INPUT-

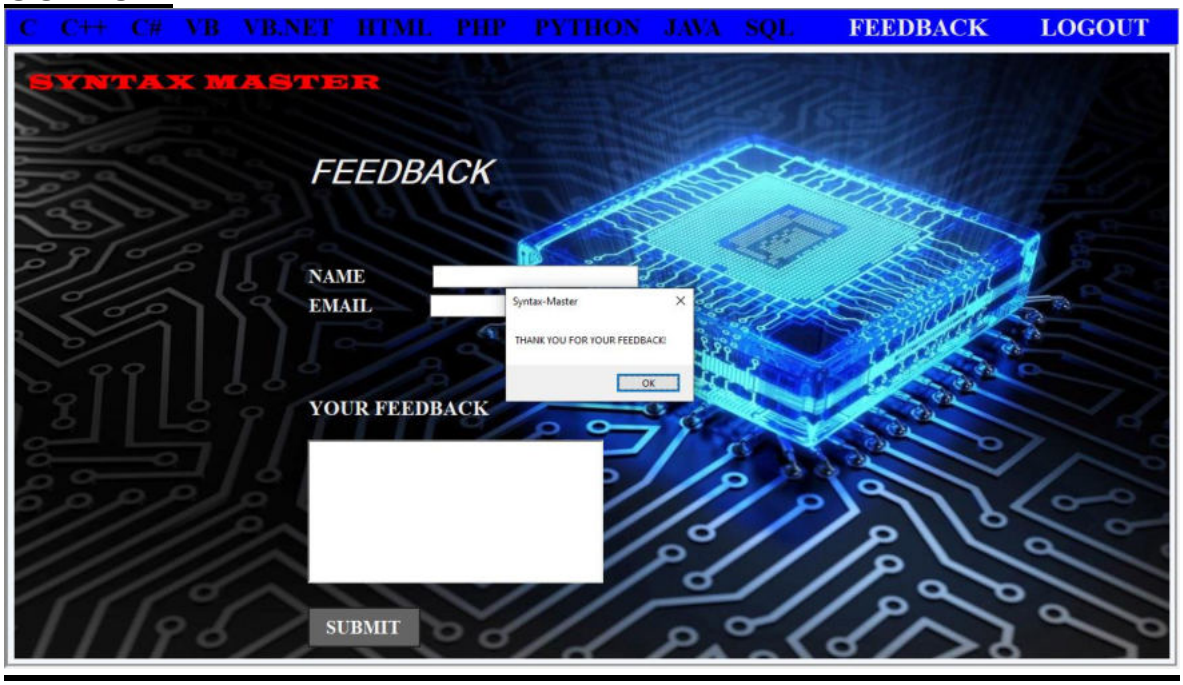
OUTPUT-

FEEDBACK FORM-

INPUT-



OUTPUT-



TESTING AND VALIDATION CHECKS

TESTING AND VALIDATION CHECKS

In this process of checking that a software system meets specifications and that it fulfils its intended purpose .It may also be referred to as software quality control. It is normally the responsibility of the software testers as part of the software development lifecycle.

Validations are nothing but the security measures taken at time to time for the execution of the program it is necessary for the analyst to set the validations in their project as it provides more accuracy and systematic flow of the application software. Validation not only stops input for the false data but also provides information in the form of message to the user to correct the mistake hence it plays the role of a guide during input of data.

Validation put its control over the data in both character as well as required data type. Whenever wrong data or invalid data is stored by the user it fires a message immediately and warn user to input correct data type.

Validations used in SYNTAX MASTER are as follows:-

1.Case Sensitiveness- First validation provided in the project is case sensitiveness the user name and the password which is generated by the admin for users and the admin are case sensitive which means the capital and the small alphabets differ.

2.Information provided is in Read Only mode-Whatever information which is provided in this software is in Read Only mode, that means no user can edit the data or information which is given in this software.

3.Password length-The Password length which is given for the users should be of minimum 8 character , in this the password length should not be less than 8 characters.

4.Value entered check- this is used for things like required fields in login forms where the user must enter some data (for example their username and password) and must not leave the field blank.

5.Permitted character check- In this it checks the characters of one field to another field (for example- Password field should match with the confirm password field)

6.Email address check- At the time of registration , the system can only accepts an email id with "@gmail.com" followed by that.

SYSTEM SECURITY MEASURES

SYSTEM SECURITY MEASURES

The objective of system security is the protection of information and property from theft, corruption and other types of damage, while allowing the information and property to remain accessible and productive. System security includes the development and implementation of security counter measures. There are a number of different approaches to system security.
In this proposed system, we have provided the following security measures-

1.Password-
The most widely method to prevent unauthorized access is to use passwords. The password needs to be kept secret and is only intended for the specific user. In this system, each password is associated with a specific username since many individuals may be accessing the same system.

2.Accessibility-
The information provided in this system for the users mode can be accessed in the read only mode which prevents the information from being edited.

3.Reports accessibility-
Reports which are generated in this system can only be accessible by the admin or the developer of this software, it is not accessible for any normal user.

IMPLEMENTATION, EVALUATION AND MAINTANANCE

IMPLEMENTATION, EVALUATION AND MAINTANANCE

Implementation

In this project the type of implementation used is conversion of manual system into computerized system which is very easy to handle and saves time which is very valuable in today's world.

System implementation involves conversion of design into the actual working system , the system implementation stands for conversion are of three types-

1.Concerning the manual system into computerised system.

2.Converting of existing system into modified version of hardware.

3.Keeping the hardware and software same and implementing the new techniques.

In the last phase of system development we have implemented the information system. This phase involves training users to handle the system some training is done by users but oversight of training is the responsibility of system analyst.

In addition the analyst need to plan for a smooth conversion from old to new system, this process includes converting file from old format to zero ones or building a database, installing equipment and bringing the new system into use.

Evaluation

We have been evaluating the evolving information system and networks to given feedback for our essential improvements. It recognition the evaluation of information system and network is important. We have taken many evolutions techniques devised. These techniques include cost-benefit analysis modules that attempts to estimate the values of a decision based on the effects of revised information using information theory, stimulation , user evaluation that emphasises implementation problem and user evaluation involvement and information system utility approaches that examine the properties of information.

The evaluation phase includes the study of existing system, there drawbacks and the various options to improve the system. The concentration should be on the satisfying the primary requirements of the users.

The system is evaluated on the basis of :

- System availability
- Compatibility
- Cost
- Performance
- Usability

In this project evaluation is made on the existing system and their drawbacks, what improvement can be made to provide facility to user. Collecting the data required for improvement in the project and then implementing it in real world. Evaluation is shown as a part of System Development Life Cycle but evaluation takes place in each and every phase.

Maintenance-

Maintenance covers a wide range of activities, including correcting, coding and design errors, updating user support. The project needs maintenance in future if any enhancement are made, maintenance of hardware and software are required as the project can be places in marked and system can handle the load.

Maintenance is an outgoing process over the life cycle of an information system. After the installation and maintenance is done, the project actually takes a form of connecting previously undetected program errors. Once these are connected the system approaches a steady state, providing dependable service to its users. The computer program must be kept up to date.

Maintenance is performed for two reasons the first reason is for the software error correction no matter how thoroughly needs to be tested the other reason is to enhance the software capability in response to changing organisational needs.

FUTURE SCOPE OF THE PROJECT

FUTURE SCOPE OF THE PROJECT

This project **"SYNTAX MASTER"** is basically a software which is developed for the students who are in a programming line. This software meets the requirement ofL the current scenario and overcomes the flaws present in the current system. But for the smooth functioning of the system the updations and modifications are necessary so that the desired result is achieved and the user of the system is satisfied.

1.More languages can be added- In this software only limited programming languages are there , but in future we can add more and more programming languages and there different topics.

2.Conversion to mobile app-Here we can convert the vb.net application to a mobile app that allows the app to be used on the go-virtually anywhere and also increases usability, functionality and feature customisation.

3.Teaching Videos-We can add the feature of teaching videos in that, users can connect to that videos and can clear their doubts regarding programming language.

4.Quiz-We can add different quizzes and question & answers for the user for their learning.

5.SMS and Email facilities- In upcoming time users can ask for their queries and requirements through email and their doubts can be cleared through mail or SMS.

6.More examples-In future we can add more examples of different topics of the particular programming languages.

CONCLUSION

CONCLUSION

Syntax master initiates the objective of providing the user with customized software of providing the basic syntax and example of all the basic programming languages . This software includes all the important and basic programming languages like C programming, C++ programming ,JAVA , HTML , C#.net , SQL etc.

All the requirements specified during the analysis and design phase are duly met , thus resulting in formation of a suitable software. The interface provided is very attractive and flexible to use.

Making this project is helpful for us to understand VB.NET language very thoroughly and using it with database by using MS ACCESS.

Excluding all the efforts made for this project there are some bugs in the system, which are still to be debugged.

it is useful for both students as well as teachers as they can seek for basic syntax and example for their learning.

Finally, the conclusion of the project is to develop the logic under the working of with the different kinds of application which are joinable with the IT profession.

We hope that the project will fulfill all its basic requirements for which it is developed there by underlying the success of this software.

BIBILIOGRPHY

BIBILIOGRPHY

WEBSITES:

- https://www.tutorialspoint.com/
- https://www.w3school.com/
- https://www.codescracker.com/

BOOKS:

- System Analysis And Design

  Author-Elias M Awad

- Vb.NET Black Book

APPROVED COPY OF SYNOPSIS

A
PROJECT SYNOPSIS
ON

"Syntax Master"

Submitted to

Rashtrasant Tukadoji Maharaj Nagpur University,
NAGPUR
In the Partial Fulfillment of

B.Com. (Computer Application) Final Year

Synopsis Submitted by
Ananya Ghanshyam Srivastava
Aniket Narhari Deshmukh

Under the Guidance of

Pravin J. Yadao

G. S. College of Commerce & Economics
Nagpur

2019-2020

1. Introduction: (Write 4 to 5 lines)

"Syntax Master" is a Windows Application which is made in VB.Net. It's central objective is to provide Syntax with examples of different programming languages like C, C++ , Java and many more languages. The main forms of this application are login form then the form which includes options for different programming languages and then the form which will provide Syntax and examples of the particular language , and after that it will also include a feedback form. This application's main motive is to provide Syntax of different programming languages in a common Application.
In this application VB.Net is used as frontend and MS Access and Notepad is been used as Backend.

2. Objectives of the project: (Write only 5 points)

a) To provide Syntax of different programming Languages in one Application.
b) For each programming language, the application contains important Syntax and their examples and output.
c) Covers all major programming languages : Python, Java, C++, C, C#.
d)Completely offline. Works without internet

3. Project Category: Windows Application

4. Tools/ Platform/ Languages to be used: VB.Net , MS Access , Notepad /Windows 10

5. Scope of future application: (Write 4 to 5 points)

a) To add more programming Languages.
b) To add more and more examples.
c) To add more description about the language or syntax.
d) To make it more dynamic.

Approved by,

Submitted by,
Ananya Ghanshyam Srivastava
Aniket Narhari Deshmukh
Name and Signature of the student

Prof. Pravin Yadao
Project Guide