|  |  |  |
| --- | --- | --- |
| C++ PROJECT-FACEBOOK | February 17  2010 | |
| C++ project on “face book”. Contents include- summary of project, Header files used, main logic behind the program, full program and output of the program after compilation. | | Compiled by-ANIRUDDH RAMRAKHYANI and ASEEM SAXENA CLASS-XI-C |

***ABOUT THE PROJECT***

Our project-‘face book’ is an initiative by **ANIRUDDH.R and** **ASSEM SAXENA** of class XI-C to create a virtual social networking module using basic techniques of c++. Though the project only focuses on creating accounts and accessing them by the use of classes, functions, header files and pre-defined functions of c++.

The code of project begins with the inclusion of header files and declaration of classes with public and private elements to be used in the creation of account. Around six different functions have been declared to be used and called in different parts of program. These include a separate independent function for the following-

* Inputting different fields for creating account
* Displaying the profile
* Displaying the home page
* Adding special effects to program like text background sounds, etc.
* Checking that user inputs correct date
* Checking that correct password is entered

The program also contains graphics like background & text colour. Another special effect includes the playing of-‘twinkle- twinkle little star’ at the beginning of program. A function in the program prints ‘\*’ at the time when password is being entered. Pre-defined functions like ‘delay’, ‘sound’, have also been used.

Many loops have been used to input password and match it with the entered while opening the profile. Structures and arrays have been used for storing characters in name, password, etc.

It has been an effort to keep the program as simple as possible with clear declaration of variables and classes. For this purpose the program has been broken down into various sub-programs and functions.

‘Face book’ is an attempt by us to bring out the clarity of understood concepts and hope we are successful in it.

***Logic behind program***

The program gives three options to user to create account/ login-account or exit from the program. For creating account a class comprising of fields such as name, age, sex, etc. has been created to store items under private and public domains. The user is asked to input various data items under suitable heads. After this information is entered the account is saved and control returns to the main screen function to display the options again. From here the option of login account can be accessed after the account has been created. On giving the option of ‘login-account’ the user is asked for e-mail id and password. The password is checked using a ‘for’ loop by comparing the ‘ith’’element of the array-‘pass’ with the entered string. If found correct the public elements of the profile are displayed using the void function-‘login’. If found incorrect the user is asked to enter password again. The same logic has been used to check email-id. The user has to enter both correctly to be able to see the public elements of his/her profile. The private elements have been kept out of the purview of the user. ‘Static int’ function has been used with ‘for’ loop for getting user’s choice. The home page is displayed by using two different functions. The home function displays the text details of the front page and also displays the text in centre and creates a box made of asterisk around the text. This has been achieved using the ‘gotoxy’ function by specifying the coordinates in parenthesis. The function-‘song’ plays out the ‘twinkle-twinkle little star’ tune before the display of home screen. ‘sound()’ and ‘delay()’ functions have been utilized for this. Textbackground() and textcolour() have been used to add background& text colour. The password function prints “\*” once a letter is typed. The getch() function doesn’t displays the letter but stores it in the designated element of array. Once the presses “enter” (ASCII code= 13/9) the loop of printing “\*” ends. For backspace “clreol” has been used. The value of ‘i’ is decremented for each backspace. If i=0, then backspace is applicable only once.

***Header files Used***

The following c++ pre-defined header files were used in the making of our project- ‘***face book’***

* Iostream.h
* Conio.h
* Process.h
* Iomanip.h
* Graphics.h
* Stdio.h
* String.h
* Dos.h