**A**

**PRACTICAL TRAINING**

**REPORT**

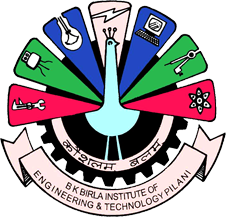
**ON**

**Student Result D**atabase System

*Submitted for the partial fulfillment of Degree*

*Bachelor of Technology*

Computer Science



**Submitted By: -** Submitted To:-

**Sanket Sharma(17EBKCS96)**

**Vishal Sharma(17EBKCS118) Mr.Rakesh saini**

**DEPARTMENT OF COMPUTER SCIENCE**

**B.K. BIRLA INSTITUTE OF ENGINEERING & TECHNOLOGY PILANI (RAJ)**

*(Affiliated to Rajasthan Technical University, Kota)*

**2018-19**

**CERTIFICATE**

**This is to certify that sanket,vishal of B.Tech CS2nd year, Roll No. 17EBKCS96,17EBKCS118. College – B K Birla Institute Engineering and Technology Pilani. Has satisfactorily carried out his practical work on Student result database system for the session 2018-2019 of part 2nd year Examination**.

**Head of Department (Dr. Nimish Kumar) Principal (Dr. L. Solanki)**

**ACKNOWLEDGEMENT**

**I would like to thank computer science department for giving me the opportunity to do a seminar presentation within a system limited. This was quite a great experience I can learn from it. It helped me to explore my skills and increased my interest in this project.**

**Special thanks to Mr. Rakesh Saini for being so accommodating and understanding. They support my field of interest in giving a presentation on student result database system . They encourage me a lot and I am very grateful to them.**

**The environment was excellent in this college and colleagues of one load make me happy every time.**

**INTRODUCTION**

Student Management System deals with all kind of student details, academic related reports, college details .It tracks all the details of a student from the day one to the end of his numbers course which can be used for all reporting purpose, tracking the numbers, progress in the student, completes assignment details, final exam result etc.

Our design can facilitate us to explore all the activities happening like exam in the college.

The student management system is an automated version of manual Student Management System. It can handle all details about a student. The details include college details, subject details, student personnel details, academic details, exam details etc.

Here almost all work is computerized. So the accuracy is maintained. Maintaining backup is very easy. It can do with in a few minutes. Our Student management system is managed by an administrator. He/she would only view details of the student. He/she can't perform any changes.

Project management

Project management skills are put to good use for this project. Having gone through project management modules in Time Series Analysis, Optimization and with two interns Project Management for Business and IT respectively, they enhanced my knowledge on managing a project. Project management focuses on achieving the objectives by applying processes presented in Figure below.



Student management system

There are records on the student result on student management

system . Through the researches, it is observed that there are features

where this project can adopt and implement. One of it will be with addition

of new result class, or even upgrading of students to the next level, the

school administrator can easily. . It can handle all details about a student.

The details include college details, subject details, student personnel

details, academic details, exam details etc.

Resources

1. System Development life cycle

Systems Development Life Cycle (SDLC) is the most common process adopted to develop a project and not surprisingly, this project is following this model too. To be precise, waterfall model is being applied. Waterfall model is a sequential model process where the input of a phase actually results from the previous phase.



There are five phases in this model and the first phase is the planning stage. The planning

stage determines the objectives of the project and whether the project should be given the

green light to proceed. This is where the proposal submission comes into picture. After

obtaining the approval, the next phase is analysis. Gathering and analysing the system anduser requirements is essential for entry to the design step. With the user requirements gathering completed, there is a need to prepare the resources for the project. Be it software or hardware components, careful consideration and selection is to be taken care at this stage. The decision on the appropriate resources to be used is further elaborated under the subsections below. The next step is to design the system and database structure.

Results from the analysis and preparation that were concluded from the previous stage are

put into action. With the user requirements in mind, the of the system is planned and

the user interface is designed to suit their easy navigation needs. In addition, the number of

tables, attributes, primary and unique keys of the database is listed.

After completing the design, actual coding begins. Database is created and codes are

written. Some of the codes required amendments and improvement to it so these are being

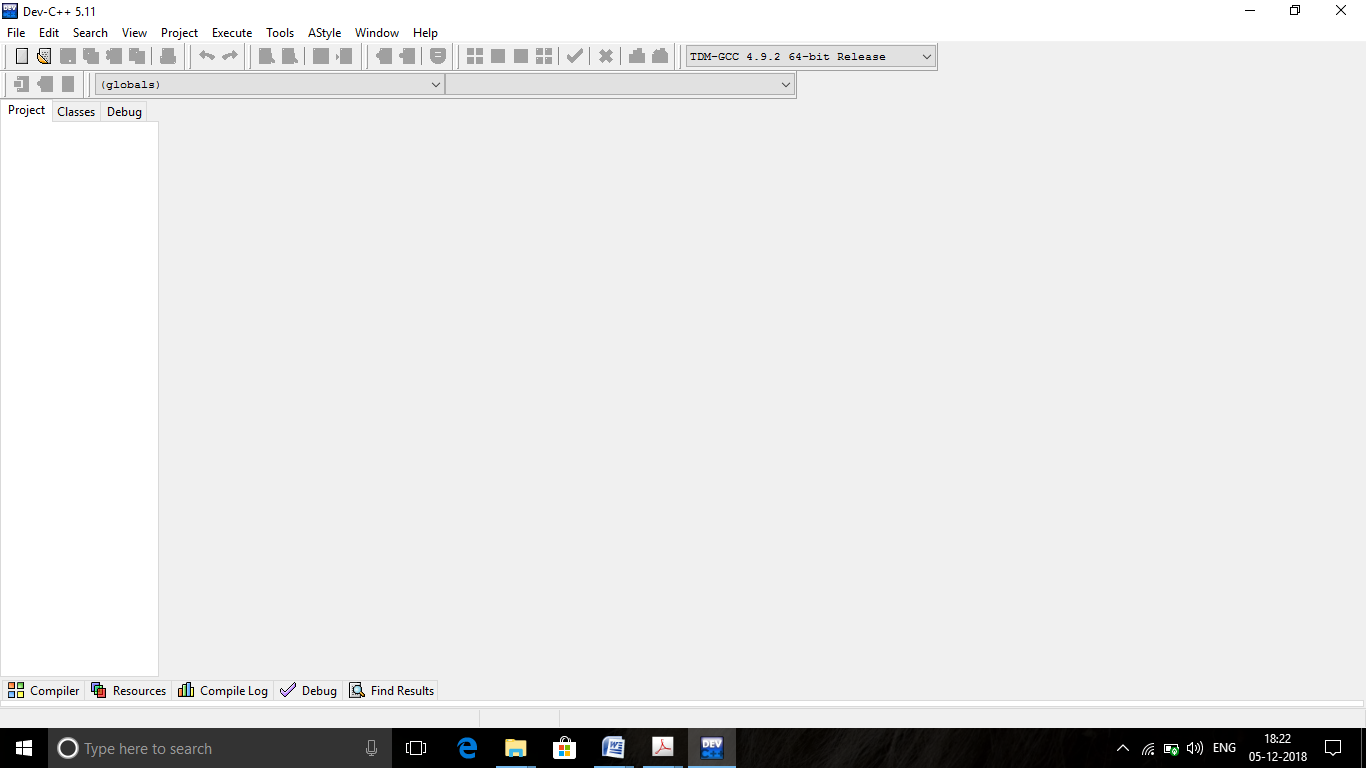
developed at this fourth stage of the waterfall model. With the development completed.

2.Scripting language selection

We select c++ language for student result database system. Because c++ language is object

Oriented programming. It is the basic language for software devloment. There history of the

C++ language. And software development int dev c++.the software devlop in c++ application.



3.History of c++

The C++ programming language has a history going back to 1979, when [Bjarne Stroustrup](http://www2.research.att.com/~bs/) was doing work for his Ph.D. thesis. One of the languages Stroustrup had the opportunity to work with was a language called Simula, which as the name implies is a language primarily designed for simulations. [The Simula 67 language](http://staff.um.edu.mt/jskl1/talk.html) - which was the variant that Stroustrup worked with - is regarded as the first language to support the object-oriented programming paradigm. Stroustrup found that this paradigm was very useful for software development, however the Simula language was far too slow for practical use.  
  
Shortly thereafter, he began work on "C with Classes", which as the name implies was meant to be a superset of the C language. His goal was to add object-oriented programming into the C language, which was and still is a language well-respected for its portability without sacrificing speed or low-level functionality. His language included [classes](http://www.cplusplus.com/doc/tutorial/classes/), basic [inheritance](http://www.cplusplus.com/doc/tutorial/inheritance/#inheritance), [inlining](http://www.cplusplus.com/doc/tutorial/functions2/#inline), [default function arguments](http://www.cplusplus.com/doc/tutorial/functions2/#default_values), and strong type checking in addition to all the features of the C language.  
  
The first C with Classes compiler was called Cfront, which was derived from a C compiler called CPre. It was a program designed to translate C with Classes code to ordinary C. A rather interesting point worth noting is that Cfront was written mostly in C with Classes, making it a self-hosting compiler (a compiler that can compile itself). Cfront would later be abandoned in 1993 after it became difficult to integrate new features into it, namely C++ [exceptions](http://www.cplusplus.com/doc/tutorial/exceptions/). Nonetheless, Cfront made a huge impact on the implementations of future compilers and on the Unix operating system.  
  
In 1983, the name of the language was changed from C with Classes to C++. The ++ operator in the C language is an operator for incrementing a variable, which gives some insight into how Stroustrup regarded the language. Many new features were added around this time, the most notable of which are [virtual functions](http://www.cplusplus.com/doc/tutorial/polymorphism/#virtual), [function overloading](http://www.cplusplus.com/doc/tutorial/functions2/#function_overload), references with the & symbol, the const keyword, and single-line comments using two forward slashes (which is a feature taken from the language BCPL).  
  
In 1985, Stroustrup's reference to the language entitled *The C++ Programming Language* was published. That same year, C++ was implemented as a commercial product. The language was not officially standardized yet, making the book a very important reference. The language was updated again in 1989 to include protected and static members, as well as inheritance from several classes.  
  
In 1990, *The Annotated C++ Reference Manual* was released. The same year, Borland's Turbo C++ compiler would be released as a commercial product. Turbo C++ added a plethora of additional libraries which would have a considerable impact on C++'s development. Although Turbo C++'s last stable release was in 2006, the compiler is still widely used.  
  
In 1998, the C++ standards committee published the first international standard for [C++ ISO/IEC 14882:1998](http://www.iso.org/iso/catalogue_detail.htm?csnumber=25845), which would be informally known as C++98. *The Annotated C++ Reference Manual* was said to be a large influence in the development of the standard. [The Standard Template Library](http://www.cplusplus.com/reference/stl/), which began its conceptual development in 1979, was also included. In 2003, the committee responded to multiple problems that were reported with their 1998 standard, and revised it accordingly. The changed language was dubbed [C++03](http://www.iso.org/iso/catalogue_detail.htm?csnumber=38110).  
  
In 2005, the C++ standards committee released a technical report (dubbed TR1) detailing various features they were planning to add to the latest C++ standard. The new standard was informally dubbed C++0x as it was expected to be released sometime before the end of the first decade. Ironically, however, the new standard would not be released until mid-2011. Several technical reports were released up until then, and some compilers began adding experimental support for the new features.  
  
In mid-2011, [the new C++ standard](http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=50372) (dubbed C++11) was finished. The [Boost library project](http://www.boost.org/) made a considerable impact on the new standard, and some of the new modules were derived directly from the corresponding Boost libraries. Some of the new features included regular expression support (details on regular expressions may be found [here](http://www.regular-expressions.info/)), a comprehensive randomization library, a new C++ time library, atomics support, a standard threading library (which up until 2011 both C and C++ were lacking), a new [for loop](http://www.cplusplus.com/doc/tutorial/control/#for) syntax providing functionality similar to foreach loops in certain other languages, the auto keyword, new container classes, better support for unions and array-initialization lists, and variadic templates

**EXTERNAL INTERFACE REQUIREMENTS**

* **USER INTERFACE :**

Each part of the user interface intends to be as user friendly as possible. The fonts and buttons used will be intended to be very fast and easy to load on software pages. The pages will be kept light in space so that it won’t take a long time for the page to load. Easy to use the software.

* **HARDWARE INTERFACE:**

Hardware requirements for student result database system.

Processor: - Pentium I or above.

RAM: - 128 MB or above.

HD: - 20 GB or above.

Operating System :Linux, Windows etc

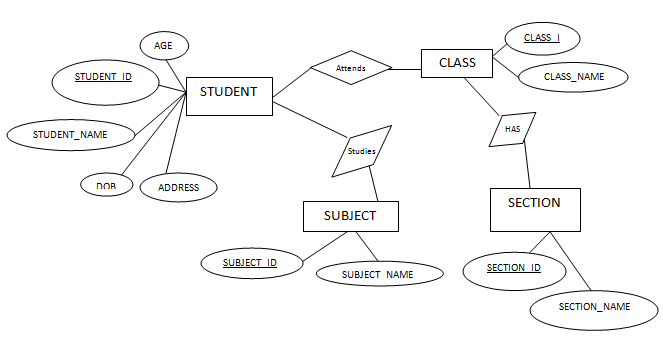
**Design**

**Er-diagram**

Student result data base system is represent by the er diagram. There are following

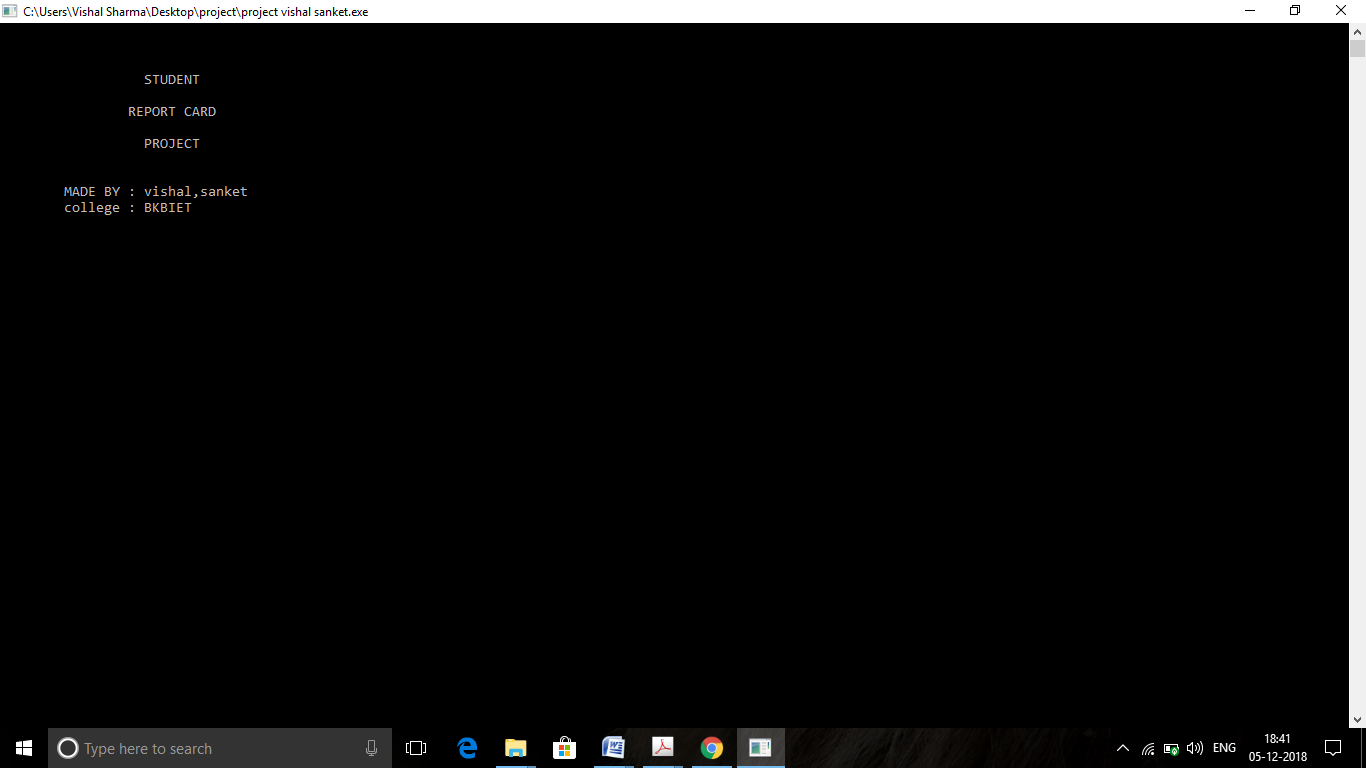
Er diagram of the project. In the er diagram basic requriment of the project basic

Working in the software.so there are the er diagram is following-

****

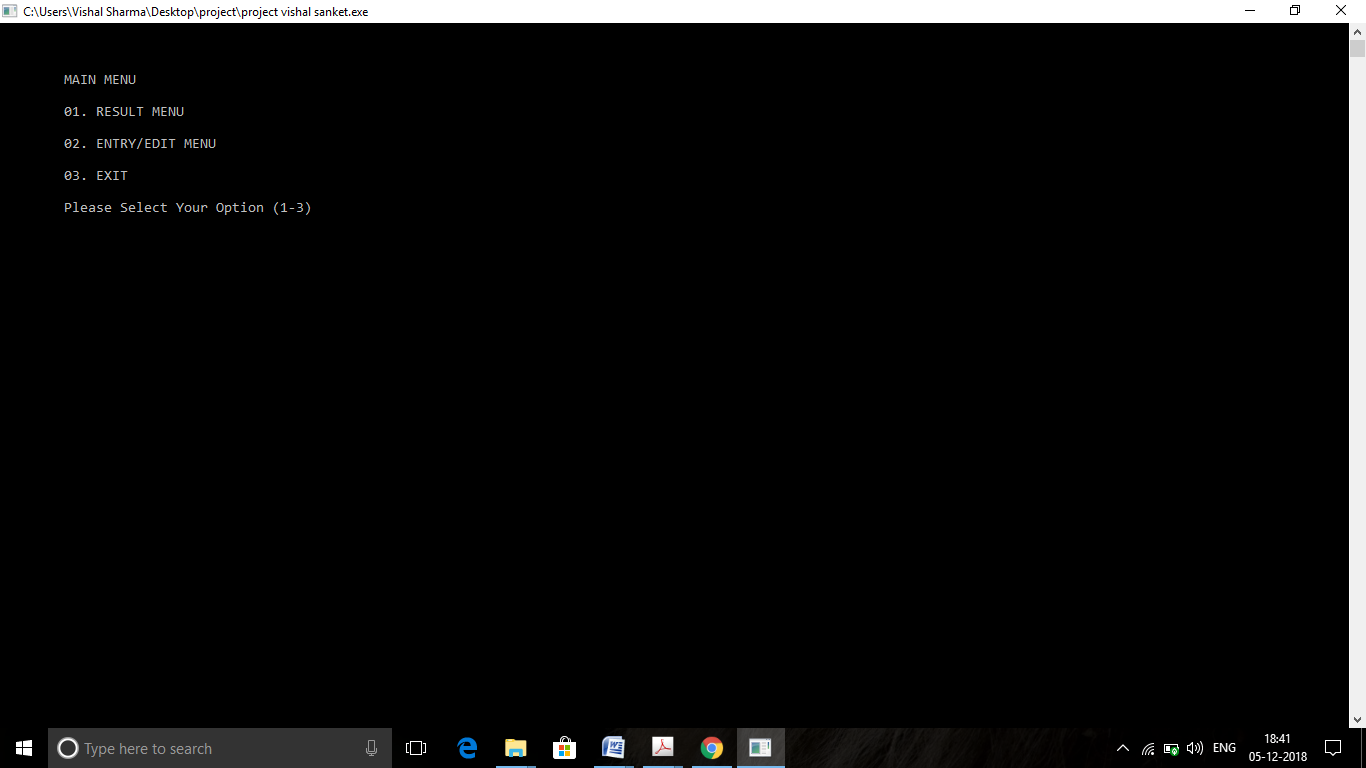
**HOME PAGE :** this is the home page of the student result result data base system.

It is the basic interface of the software.



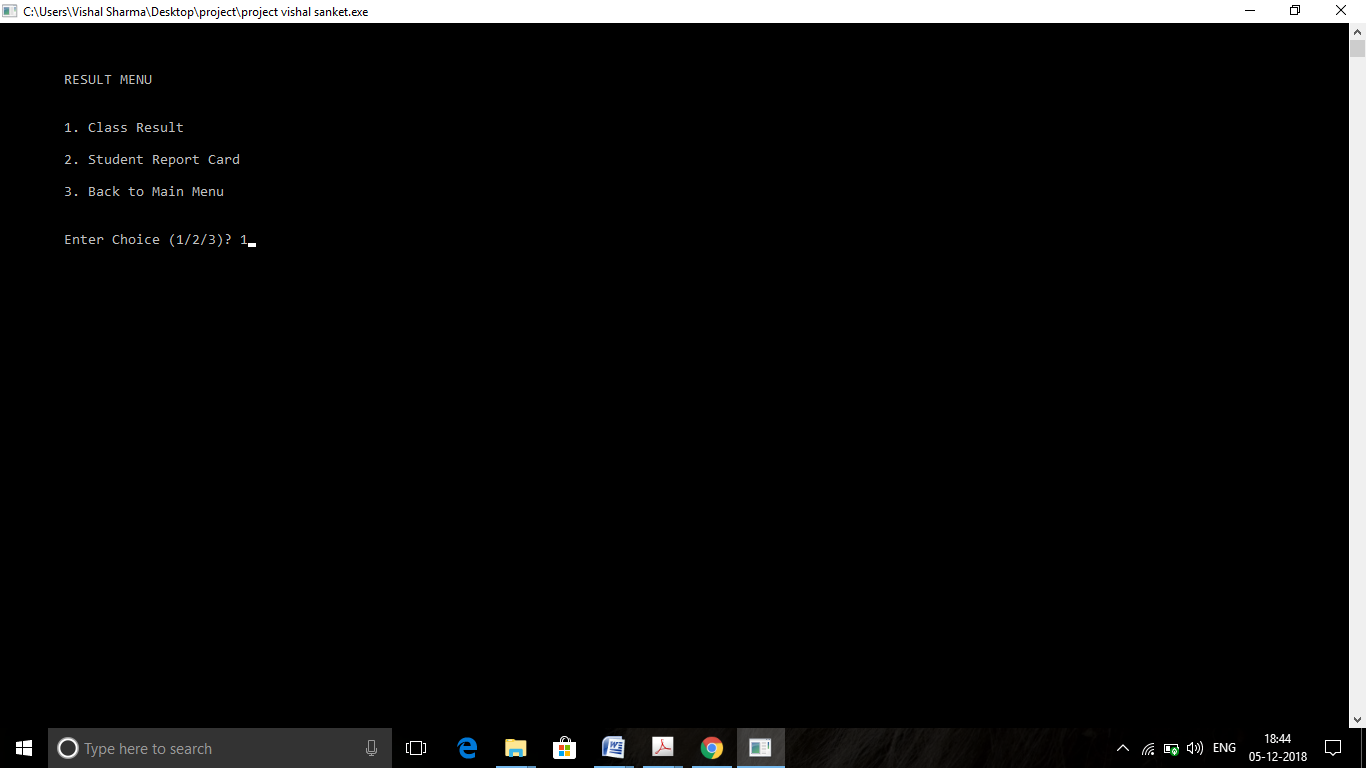
Main menu: this is the main menu of the project . in the main menu 3 functions are in

The main menu result menu, enter/edit menu, exit. so this is the main menu of the software



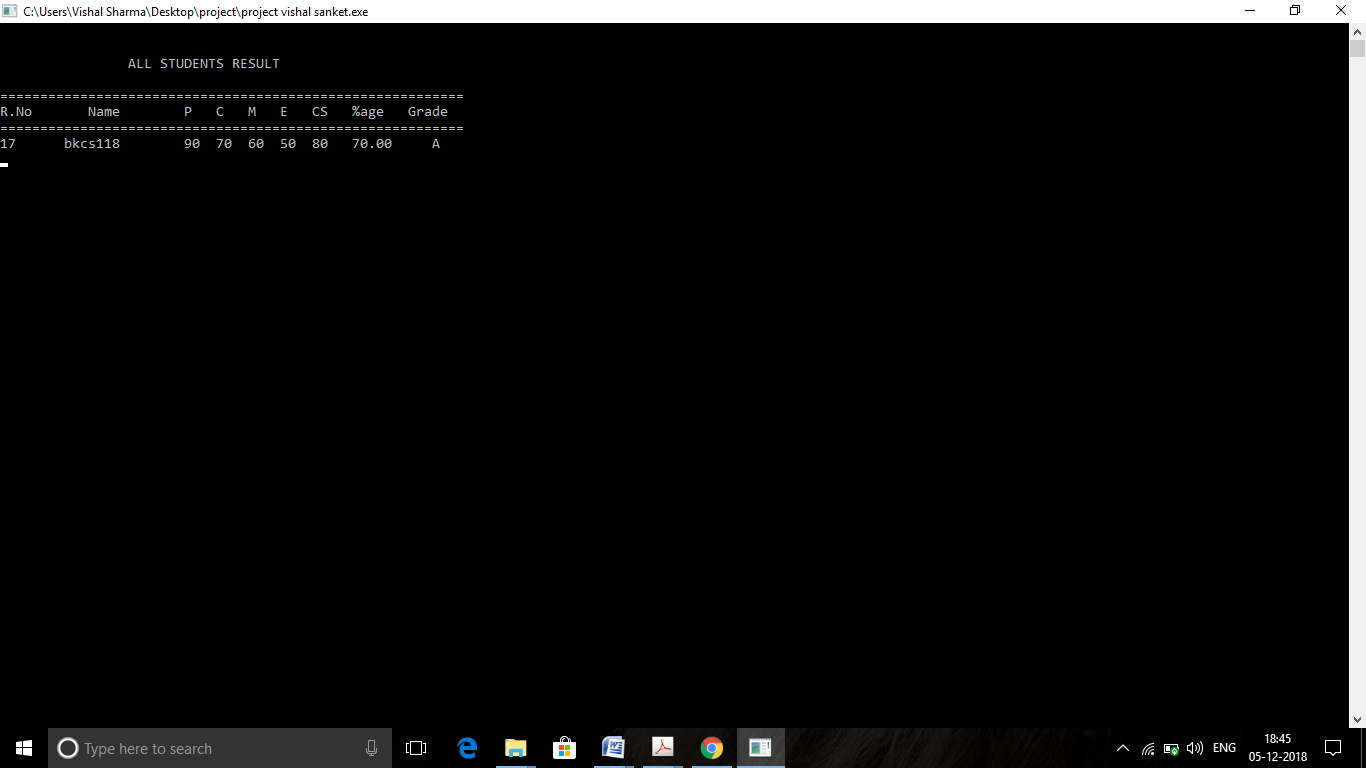
**Result menu:** The second interface of the software is class result and student report

Card and back to main menu provide in the result menu.



**Class result:**

This is the all student class result is provides. in the class result function.

****

**Enter menu:**

In the enter menu provide six function in enter menu. First function is create any student result means create record of student. And second function is the display all student

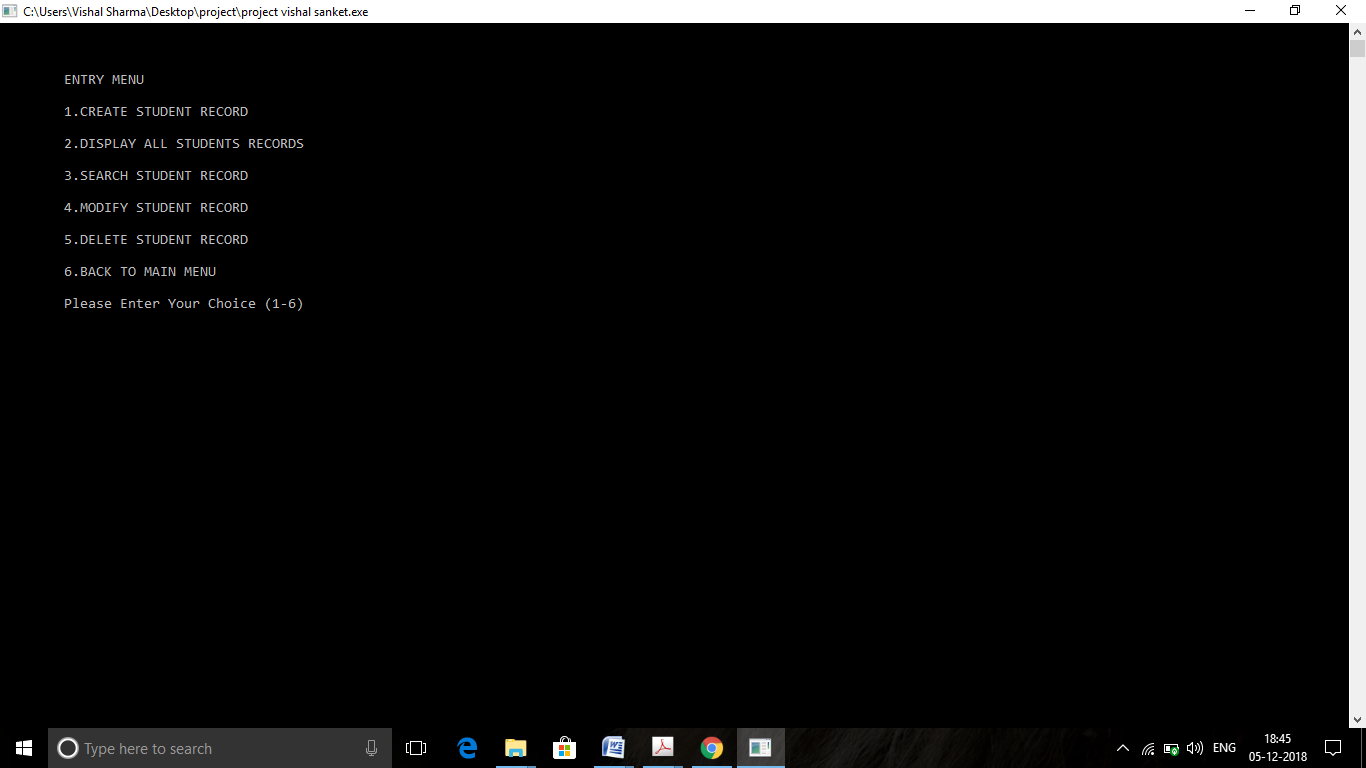
Result and show on the consol on the screen. and third function is search any student

By his roll no . means search any student result in all student result. And fourth function

Use for modify any student result record. If any detail are wrong so modify student

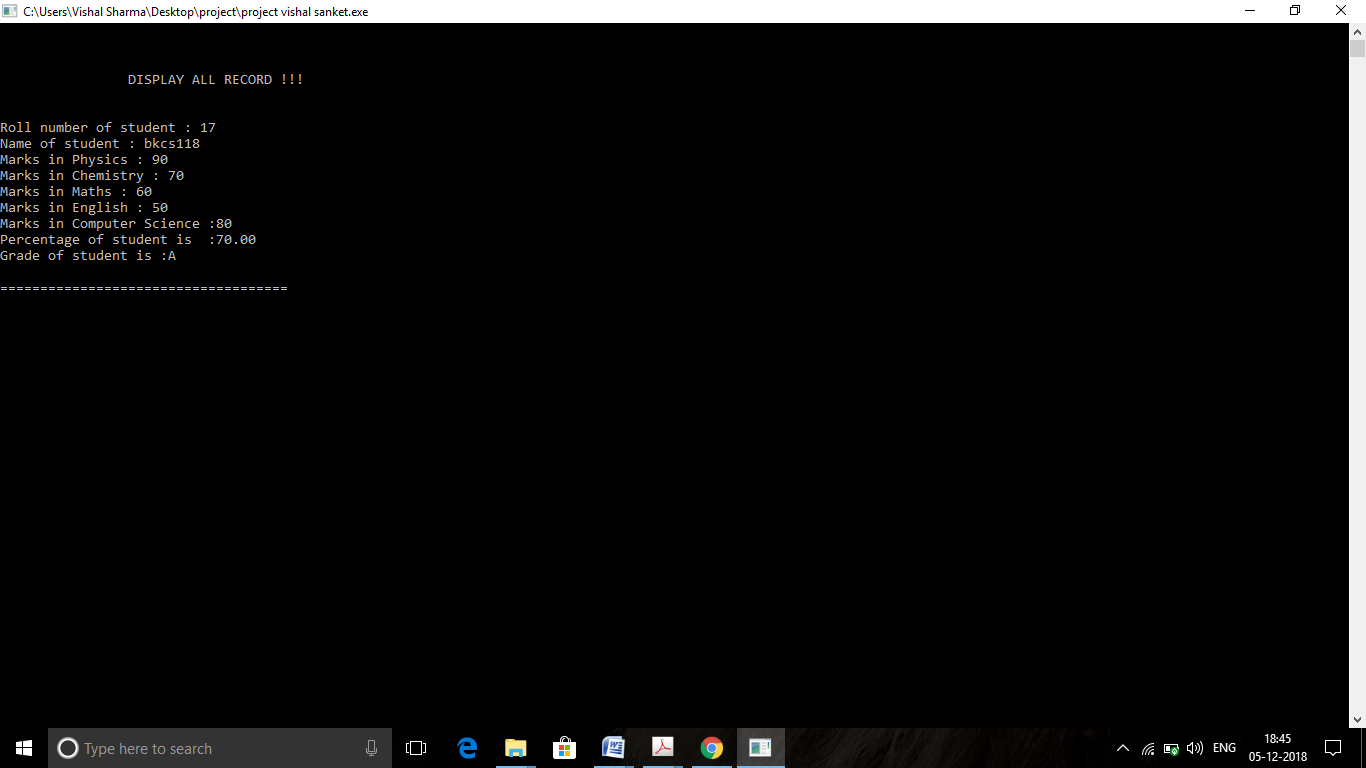
Result. And last function in the entry menu is back to main .if use this function so back

In the main menu. So these are all function in the entry menu.

****

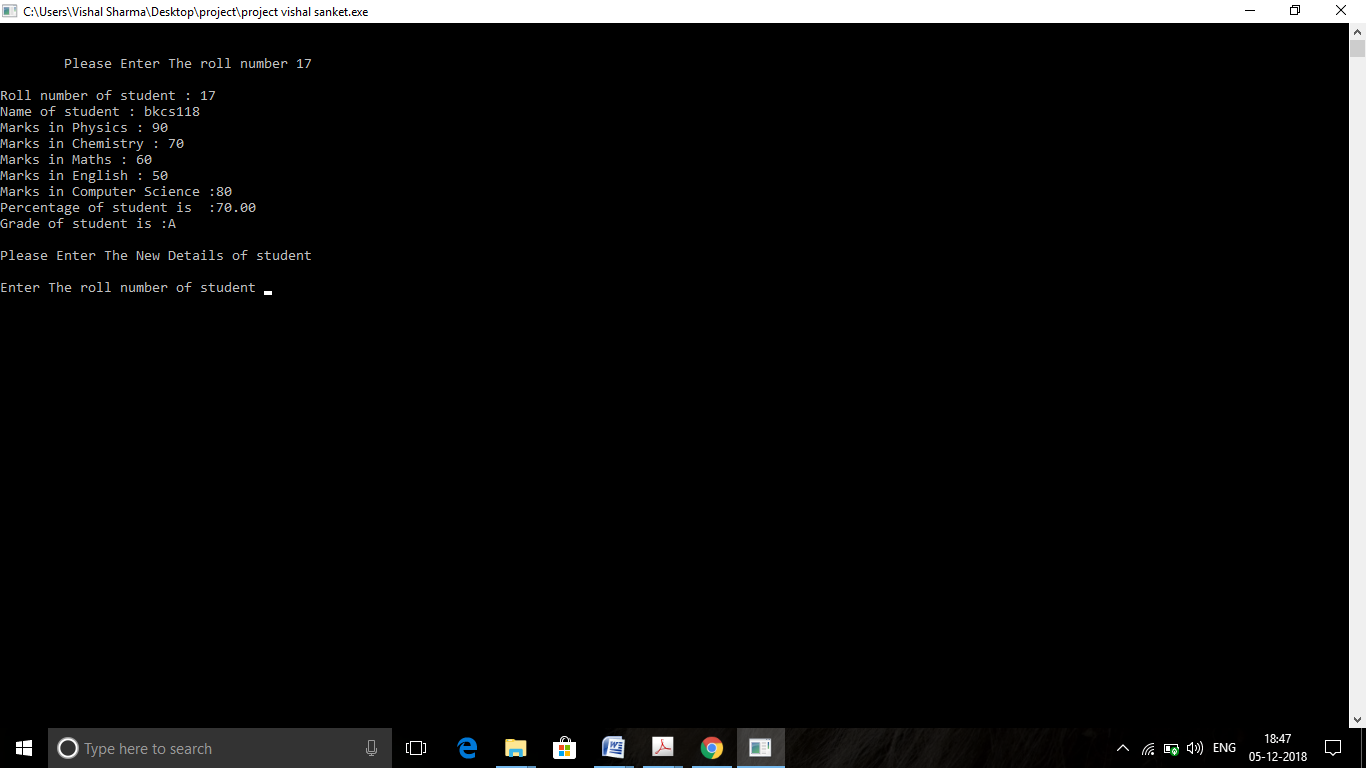
**Display student result:**

Display all result in a single function.

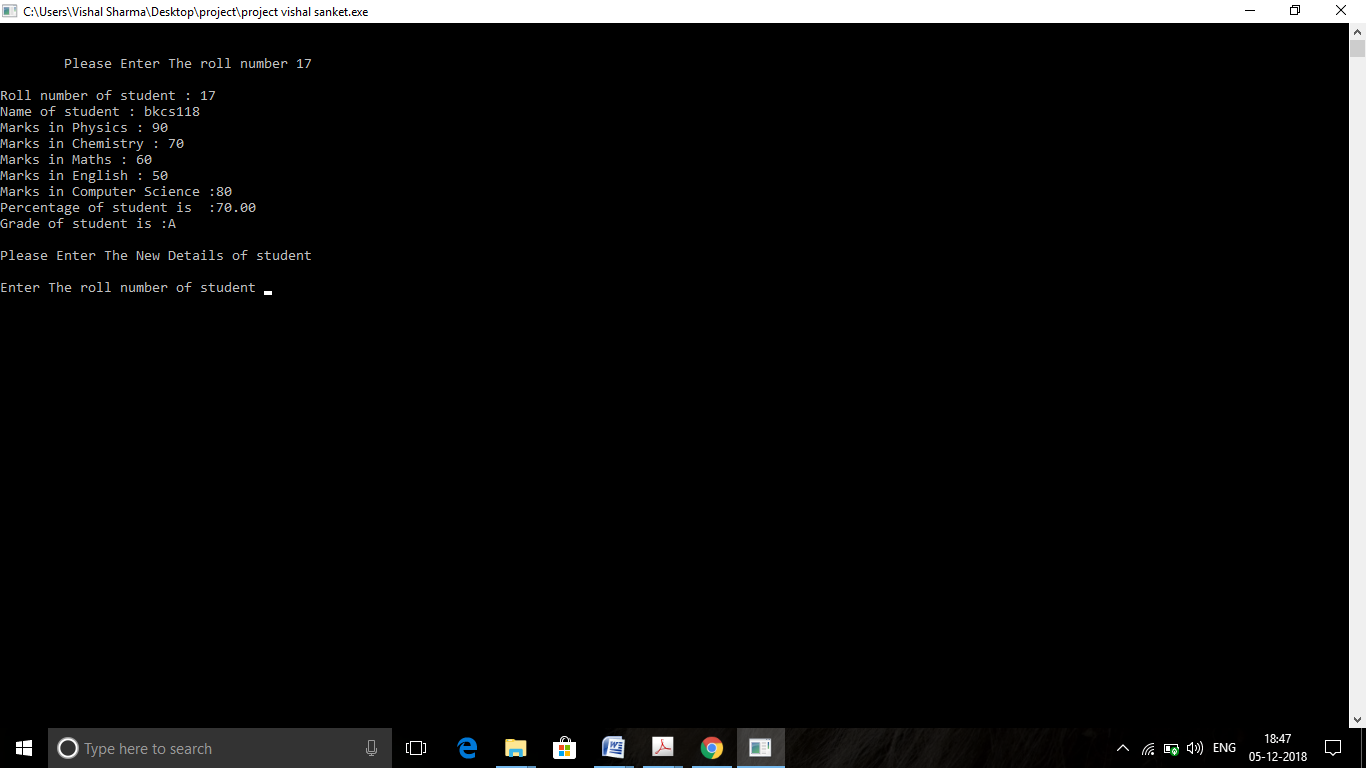


**Search result:**

search any student result by his roll number.



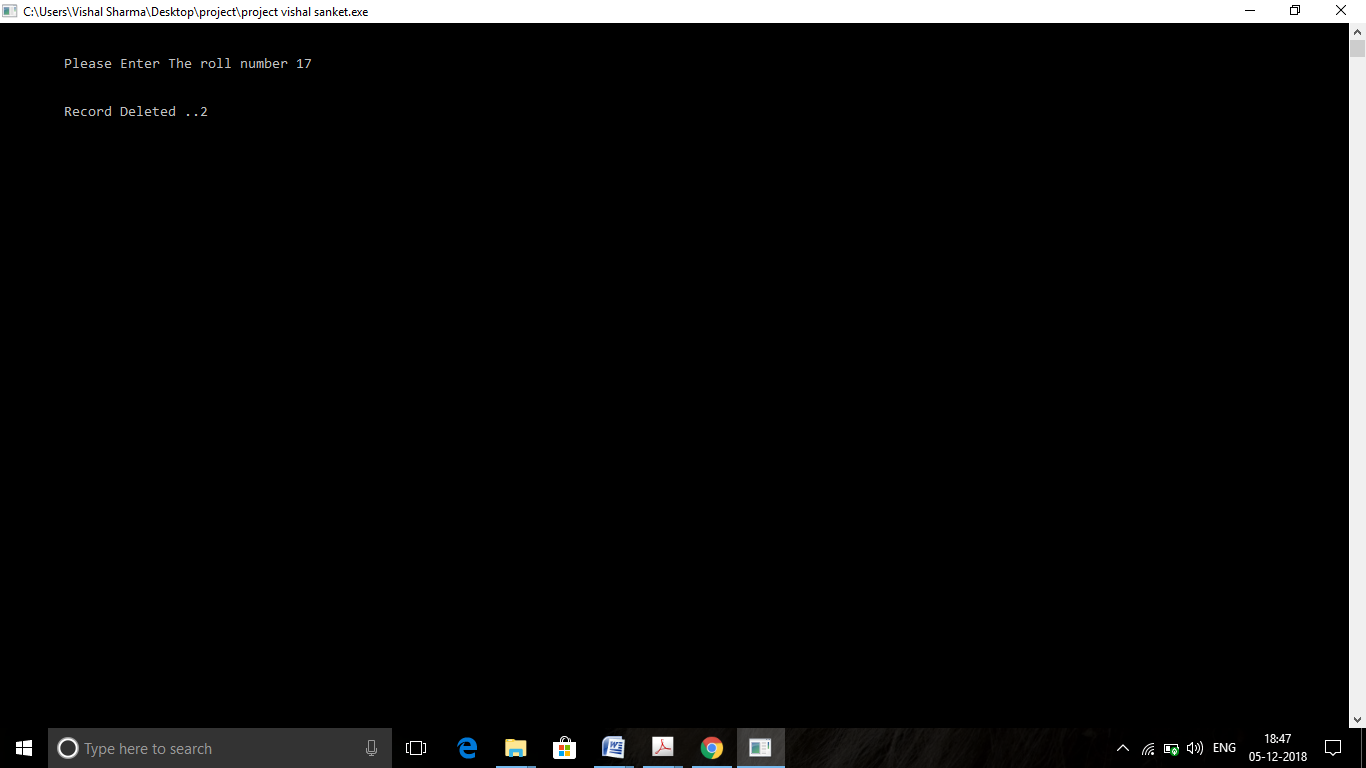
**Modify result:**

****

**Delete record:**

Using this function delete any student result record by use this function in the one

Step using this function.

****

**Source of development dev c++**

**Dev-C++** is a [free](https://en.wikipedia.org/wiki/Free_software) full-featured [integrated development environment](https://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) distributed under the [GNU General Public License](https://en.wikipedia.org/wiki/GNU_General_Public_License) for programming in [C](https://en.wikipedia.org/wiki/C_(programming_language)) and [C++](https://en.wikipedia.org/wiki/C%2B%2B). It is written in [Delphi](https://en.wikipedia.org/wiki/Delphi_(programming_language)).

It is bundled with, and uses, the [MinGW](https://en.wikipedia.org/wiki/MinGW) or [TDM-GCC](https://en.wikipedia.org/wiki/TDM-GCC) 64bit port of the [GCC](https://en.wikipedia.org/wiki/GNU_Compiler_Collection) as its compiler. Dev-C++ can also be used in combination with [Cygwin](https://en.wikipedia.org/wiki/Cygwin) or any other GCC-based compiler.[[1]](https://en.wikipedia.org/wiki/Dev-C%2B%2B#cite_note-1)

Dev-C++ is generally considered a [Windows-only](https://en.wikipedia.org/wiki/Microsoft_Windows) program, but there are attempts to create a Linux version: header files and path delimiters are switchable between platforms.

From February 22, 2005 to June 2011 the project was not noticeably active, with no news posted nor any updated versions released. In a 2006 forum post, lead developer Colin Laplace stated that he was busy with real-life issues and did not have time to continue development of Dev-C++.[[2]](https://en.wikipedia.org/wiki/Dev-C%2B%2B#cite_note-2)

There are two forks of Dev-C++ since then: [wxDev-C++](https://en.wikipedia.org/w/index.php?title=WxDev-C%2B%2B&action=edit&redlink=1) and the *Orwell* version.

wxDev-C++ is a [development team](http://wxdsgn.sourceforge.net/?q=node/19) that has taken Dev-C++ and added new features such as support for multiple compilers and a RAD designer for [wxWidgets](https://en.wikipedia.org/wiki/WxWidgets) applications.

On June 30, 2011 an unofficial version 4.9.9.3 of Dev-C++ was released by Orwell (Johan Mes), an independent programmer,[[3]](https://en.wikipedia.org/wiki/Dev-C%2B%2B#cite_note-3) featuring the more recent GCC 4.5.2 compiler, Windows' SDK resources (Win32 and D3D), numerous bugfixes, and improved stability. On August 27, after five years of officially being in a beta stage, version 5.0 was released.[[4]](https://en.wikipedia.org/wiki/Dev-C%2B%2B#cite_note-4)This version also has its own separate SourceForge[[5]](https://en.wikipedia.org/wiki/Dev-C%2B%2B#cite_note-5) page since version 5.0.0.5, because the old developer isn't responding to combining requests. On July 2014, Orwell Dev-C++ 5.7.1 was released featuring the more recent [GCC](https://en.wikipedia.org/wiki/GNU_Compiler_Collection) 4.8.1 which supports [C++1](https://en.wikipedia.org/wiki/C%2B%2B11)

**source code:**

Develop the project in the c++ language .development the student result data base system

In the c++ programming .devc++ is provide the features of c and c++ programming compile

It is the line of code in the dev c++. It is very useful for the basic data base programming.

It is use for the compaile the source code and after run the software and implementation

The source code .so this the source code .there are some figures-

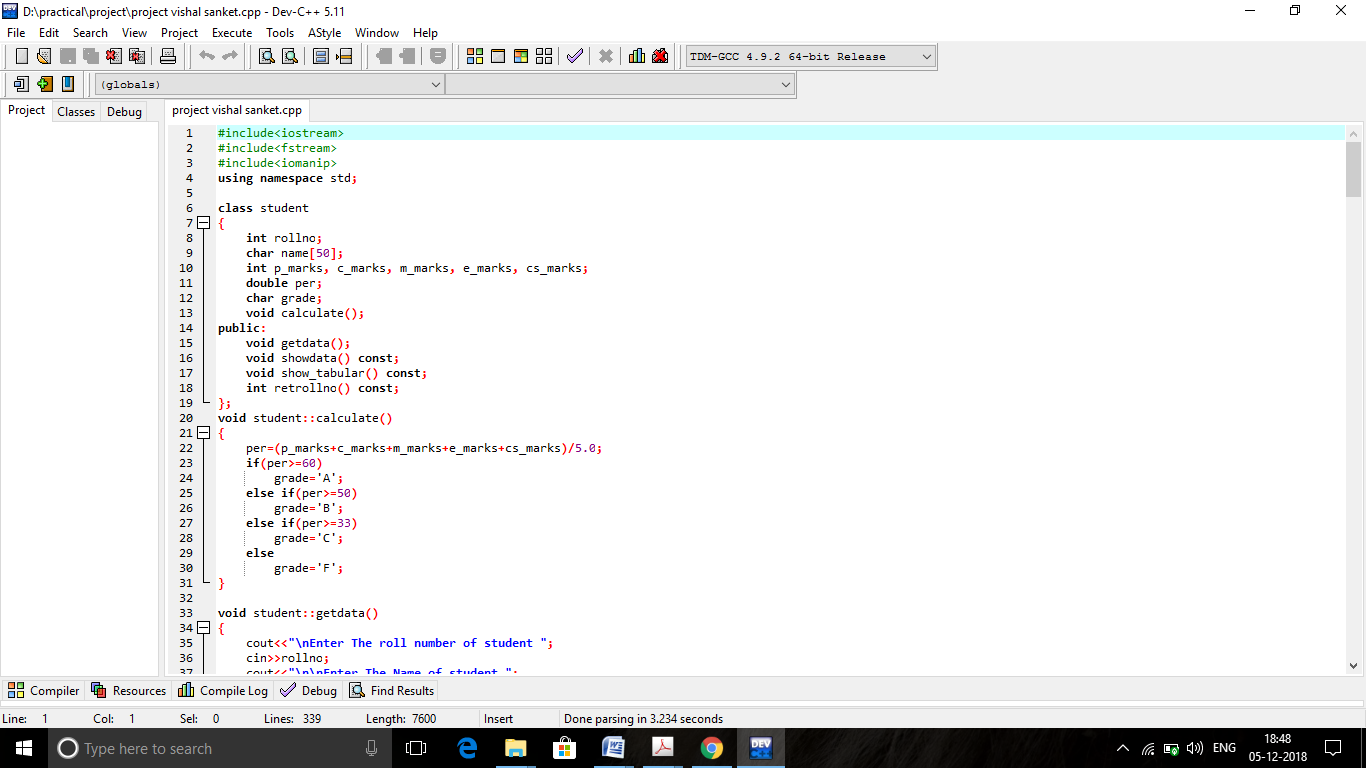
****

Figure-1

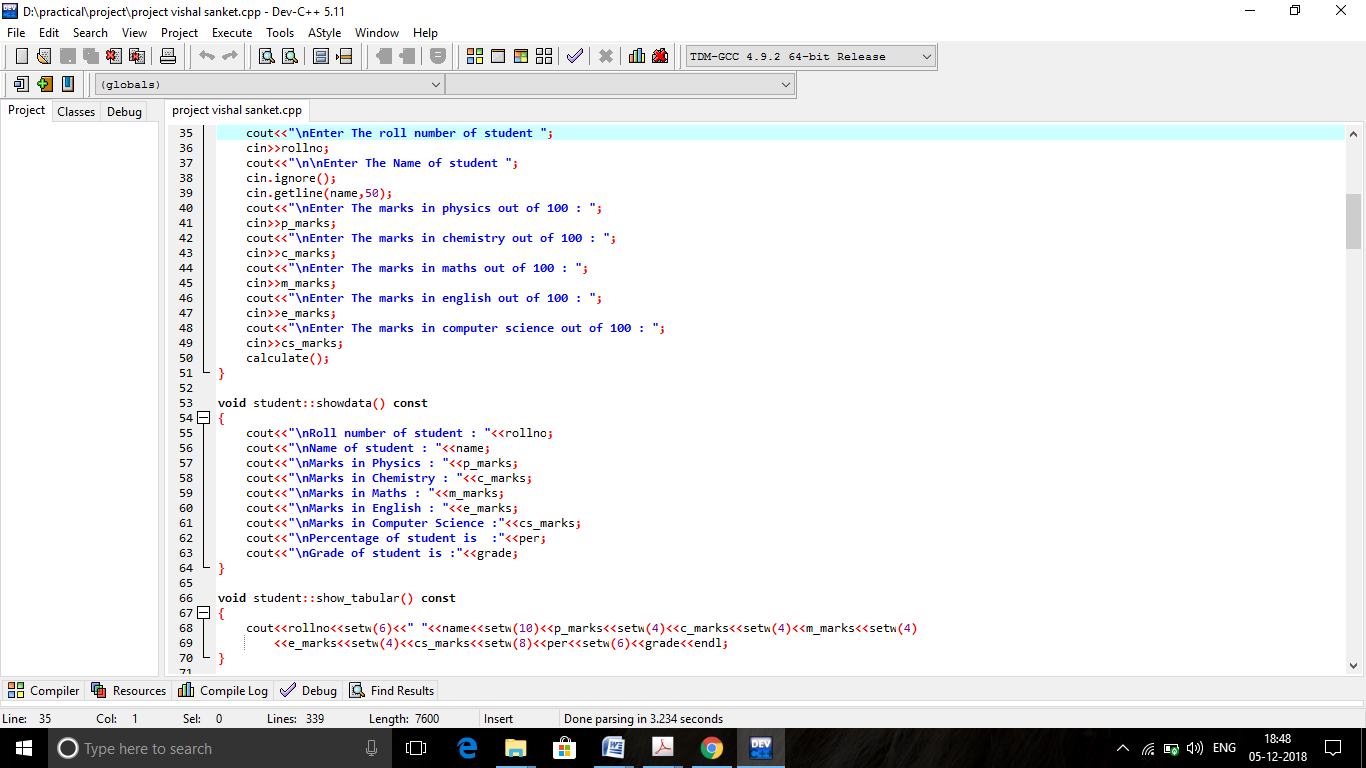
****

Figure-2

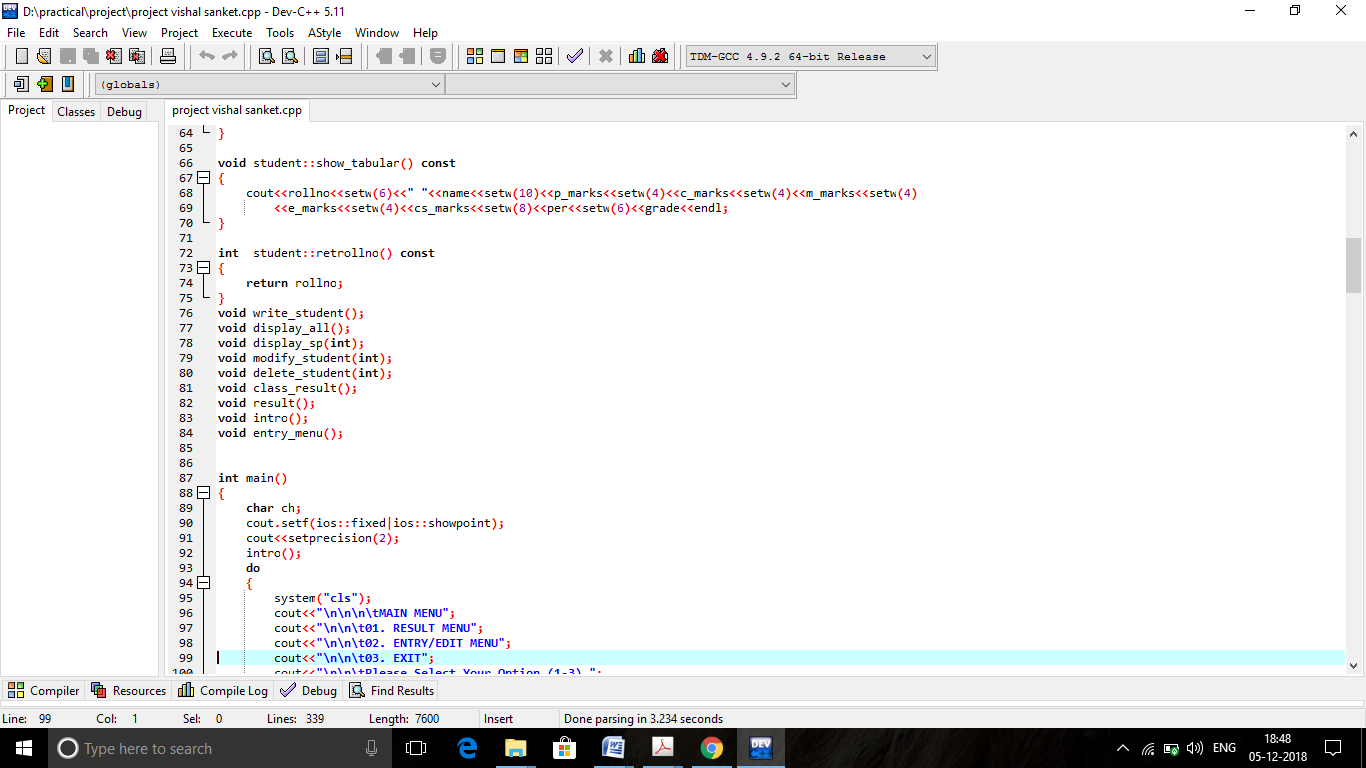
****

Figure - 3

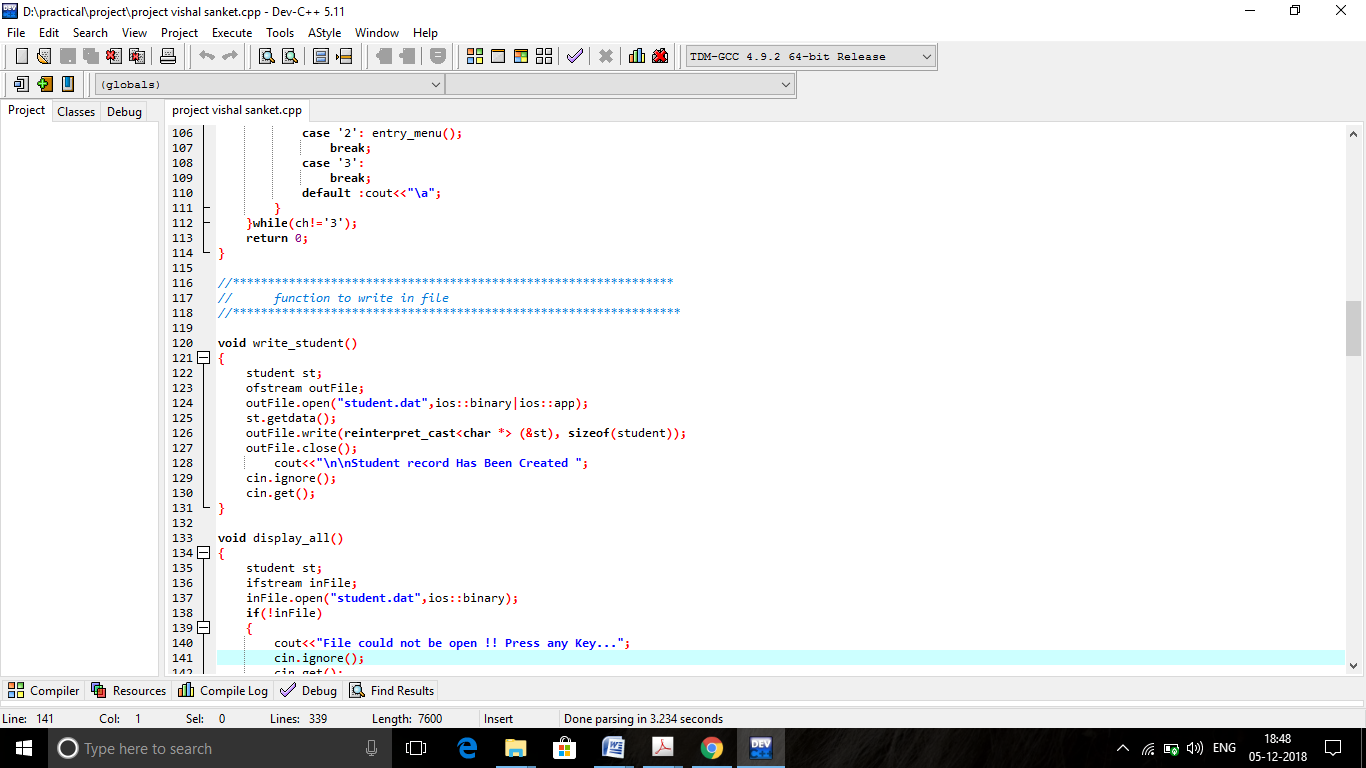
****

Figure-4

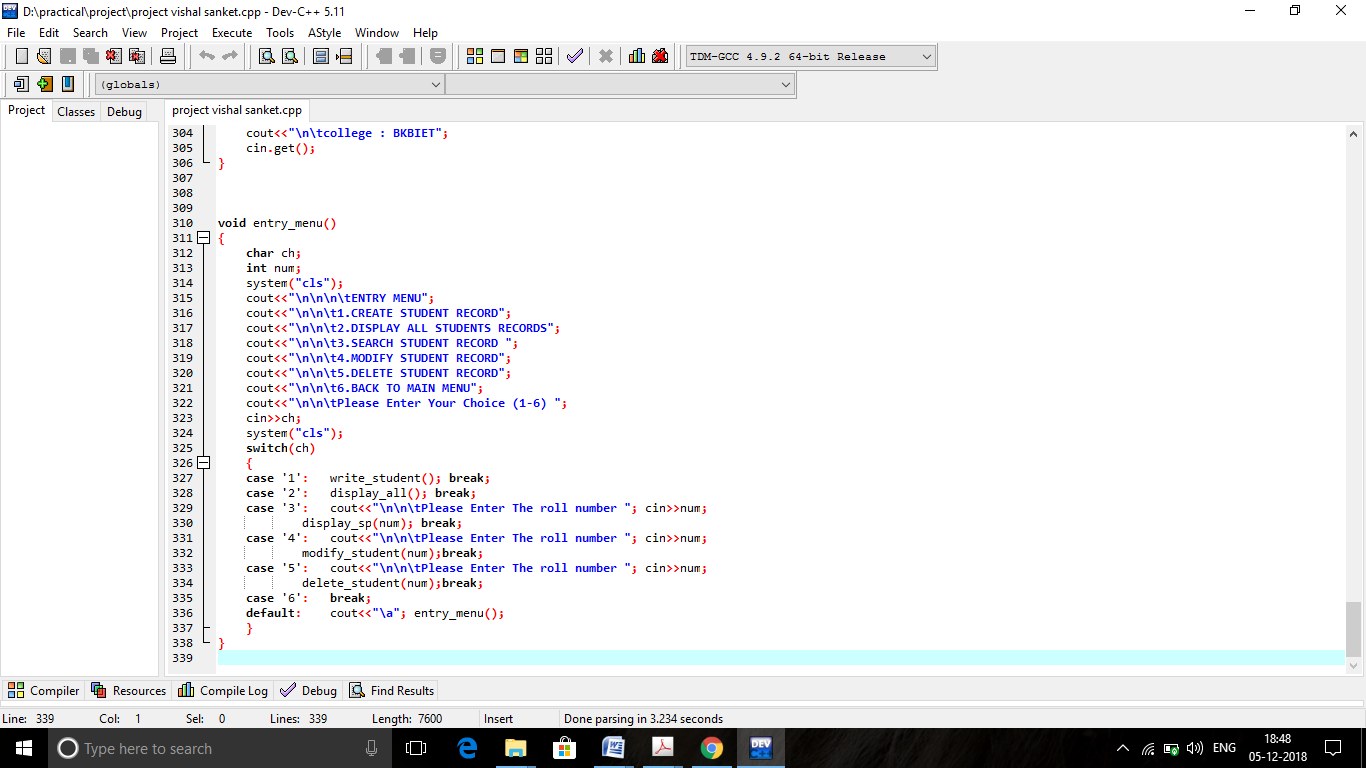
****

Figure-5

**Testing & debugging**

Testing is the last phase of this project. This is the stage where it shows whether the system

and database are working as desired. Otherwise, further rectification is required.After the system development is completed, the final step is debugging and testing. For example, in the background the administrator could set a main category and subcategory. Administrator

can add any record of the student. And check its works and not it is testing of the software

and testing of the software again because for quality of the software. The process of testing of an integrated hardware and software system to verify that the system meets its specified requirements. Verification: Confirmation by examination and provisions of objective evidence that specified requirements have been fulfilled. To test the system as a whole, requirements and expectations should be clear and the tester needs to understand real time usage of application too.

**INSTALLATION OF COMPUTING PLATFORM**

Install proposed system to run project. In this system windows platform is required. So first of all install windows 98/2000/XP/8/10 operating system.

**TECHNOLOGY TESTING**

Install sufficient hardware such as monitor, printer, keyboard etc. on site. Help also can provide to user to meet record database of the students.

**PROGRAM TESTING**

After the modules were tested & integrated with software packages both valid & invalid test transaction were run to test software system.

**INPUT TESTING**

User will be tested to determine if they are completing the forms correctly, accuracy & speed of data entry personnel will be evaluate the efficiency of screen input layout will be evaluated. inputs are provide by user how to give input and check out the input testing.

**OUTPUT TESTING**

View & enquiry screen will be tested for accuracy based on live data is entered during the training sessions. All reports will be issued to the final users, & each user will be required to fill out a questionnaire if the view report meets his/her information needs.

**DATABASE TESTING**

After live data is entered, data files are tested for completeness by comparing batch control total to produce from the database. Other database tests are performed during development.

**RECOVERABILITY TESTING**

To make sure how well the system recovers from various input errors and other failure situations.

**PERFOMANCE TESTING**

To make sure that functionality of product are working as per the requirements defined, within the capabilities of the system.

**INTROPERABILITY TESTING**

To make sure wheather the system can operate well with third party product or not.

**RELIABILITY TESTING**

To make sure system can be operated for longer duration without developing failures.

**DOCUMENTATION TESTING**

To make sure that the system’s user guide and other help topics documentation are correct and usable.

**TEST CASES**

Following of some test cases are made during entire level of testing:

**AUTHENTICATION TEST**

Verification of permitted users is done by entering the valid & authorized user’s entry on home page.

**QUERY TEST**

Testing for various queries generated in the application were tested whether query request for local database returns the correct dataset for the corresponding query as well as queries for updating (edit), save the data properly in database.

**VALIDATION TEST**

Here it was tested that, if correct or incorrect entries by the user are accepted & the processed data outputs the expected results.

**Conclusions**

Simplicity is never simple. As we have seen in this project, the process of creating a user-

friendly and straightforward platform that facilitates the administrator's job is one filed

with complexity. From understanding user requirements to system design and finally system

prototype and finalization, every step requires in-depth understanding and commitment

towards achieving the objectives of the project.

Although the student database management module is not fully integrated to the system

and used on real time, the system prototype demonstrates easy navigation and data are

stored in a systematic way. Overall, efficiency has improved and work processes simplified.

Working of the student result data base system. this is the conclusions of student result database system.

**REFFERENCE**

* **Websites**
* http://en.wikipedia.org/wiki/Online\_shopping#Advantages
* <http://www.google.com>

https://sourceforge.net

* **Book**

1.Object Oriented Programming with C++

2. The C++ Programming Language by [Bjarne Stroustrup](https://www.amazon.com/Bjarne-Stroustrup/e/B000AQ349S/ref=dp_byline_cont_book_1).

**THANK YOU**