**PROJECT ON**

**“QUIZ MASTER”**

**SUBMITTED TO**

**CENTRAL BOARD OF SECONDARY EDUCATION**

**COMPUTER SCIENCE**

**(CLASS XII)**

**2021-2022**



Submitted By:

**SAIKAT SAHA** (AISSCE R.No.20611517)

**NAVY CHILDREN SCHOOL**

**VISAKHAPATNAM**

**CERTIFICATE**

This is to certify that **Saikat Saha** of **class XII**, bearing Roll No. **20611517** has successfully completed his/her project on “**QUIZ MASTER**” in database connectivity for All India Senior School Certificate Examination (AISSCE) 2022.

**Date:**

**Internal Examiner**  **Principal**

**External Examiner**

**ACKNOWLEDGEMENT**

I would like to express my profound sense of gratitude to my guide

**Mr.Mukesh bohra ,**PGT Computer Science, **Navy Children School Nausena Baugh,** for inspiring and guiding me throughout the record. The suggestions and ideas given by my guide were most valuable I came to know about so many new things.

Apart from the efforts of me, the success of any project depends largely on the encouragement and guidelines of many others. I take this opportunity to express my gratitude to the people who have been instrumental in the successful completion of this project.

I express deep sense of gratitude to almighty God for giving me strength for the successful completion of the project.

I would like to thank **Dr. Parul Kumar, Principal of Navy Children School, Nausena Baugh,** for supporting me in my endeavors.

**INDEX**

|  |  |  |
| --- | --- | --- |
| **Serial No** | **TOPIC** | **Page No.** |
| 1 | Objective of project | 05 |
| 2 | Introduction | 05 |
| 3 | Proposed System | 05 |
| 4 | System Development Life Cycle (SDLC) | 09 |
| 5 | Phases of system development life cycle | 10 |
| 6 | Flow chart | 14 |
| 7 | Source Code | 15 |
| 8 | Output | 23 |
| 9 | Testing | 34 |
| 10 | Hardware and Software  requirements | 36 |
| 11 | Biblography | 37 |

**OBJECTIVE OF THE PROJECT**

The objective of this project is to let the students apply the programming knowledge into a real- world situation/problem and exposed the students how programming skills helps in developing a good software.

1. Write programs utilizing modern software tools.

2. Apply object oriented programming principles effectively when developing small to medium sized projects.

3. Write effective procedural code to solve small to medium sized problems.

4. Students will demonstrate a breadth of knowledge in computer science, as exemplified in the areas of systems, theory and software development.

5. Students will demonstrate ability to conduct a research or applied computer Science project, requiring writing and presentation skills which exemplify scholarly style in computer science.

**INTRODUCTION**

1. This project name is Quiz Master, in which the userwill register with their details and then login to take the quiz.

1. The admin with the admin password can use this code to change the information in backend database.
2. The project is built in Python programming language as the front end and MySQL database is used at the backend.

**PROPOSED SYSTEM**

Today one cannot afford to rely on the fallible human beings of be really wants to stand against merciless competition where not to wise saying “to err is human” no longer valid, it’s outdated to rationalize your mistake. So, to keep pace with time, to bring about the best result without malfunctioning and greater efficiency so to replace the unending heaps of flies with a much more sophisticated hard disk of the computer.One has to use the data management software. Software has been an ascent in atomization of various organizations. Many software products working are now in markets, which have helped in making the organizations work easier and efficiently. Datamanagement initially had to maintain a lot of ledgers and a lot of paperwork has to be done but now software production this organization has made their work faster andeasier. Now only this software has to be loaded on the computer and work can be done.This prevents a lot of time and money.

**About Python**

Python is developed under an OSI-approved open source license, making it freely usable and distributable, even for commercial use. Python's license is administered by the Python Software Foundation. Python is a widely-used, interpreted, object-oriented, and high-level programming language with dynamic semantics, used for general Purpose programming. It was created by Guido van Rossum, and first released on February 20, 1991. While you may know the python as a large snake, the name of the Python programming language comes from an old BBC television comedy sketch series called Monty Python’s Flying Circus.One of the amazing features of Python is the fact that it is actually one person’s work. Python is an exception Of course, van Rossam did not develop and evolve all the Python components himself. The speedwith which Python has spread around the world is a result of the continuous work of thousandsprogrammers, testers, users (many of them aren’t IT specialists) and enthusiasts, but it must be said that the very first idea (the seed from which Python sprouted) came to one head – Guido’s.

**Why Python?**

What makes Python so special? How does it happen that programmers, young and old, experienced and novice, want to use it? How did it happen that large companies adopted Python and implemented their flagship products using it?

There are many reasons – we’ve listed some of them already, but let’s enumerate them again in a more practical manner:

1. it’s easy to learn – the time needed to learn Python is shorter than for many other languages; this means that it’s possible to start the actual programming faster;   
   2. it’s easy to teach – the teaching workload is smaller than that needed by other languages; this means that the teacher can put more emphasis on general (language-independent) programming techniques, not wasting energy on exotic tricks, strange exceptions and incomprehensible rules;   
   3. it’s easy to use for writing new software – it’s often possible to write code faster when using Python;it’s easy to understand – it’s also often easier to understand someone else’s code faster if it is written in Python;
2. it’s easy to obtain, install and deploy – Python is free, open and multiplatform; not all languages can boast that.

**Python in Action**  
Where can we see Python in action?   
We see it every day and almost everywhere. It’s used extensively to implement complex Internet services like search engines, cloud storage and tools, social media and so on. Whenever you use any of these services, you are actually very close to Python, although you wouldn’t know it. Many developing tools are implemented in Python.More and more everyday use applications are being written in Python. Lots of scientists have abandoned expensive proprietary tools and switched to Python. Lots of IT project testers have started using Python to carry out repeatable test procedures.

**Python Examples**

Python is a great choice for:

• Web and Internet development (e.g., Django and Pyramid frameworks, Flask and Bottle micro frameworks)

• Scientific and numeric computing (e.g., SciPy – a collection of packages for the purposes of mathematics, science, and engineering;

• Desktop GUIs (e.g., wxWidgets, Kivy, Qt)

• Software Development (build control,

management, and testing – Scons, Buildbot,

Apache Gump, Roundup, Trac)

• Business applications (ERP and e-commerce

systems – Odoo, Tryton)

• Games (e.g., Battlefield series, Sid Meier\’s

Civilization IV...), websites and services (e.g.,

Dropbox, UBER, Pinterest, BuzzFeed...)

**About mysql-connector**

A database can be accessed from within a Python program with the help of DB API (Database Application Programming Interface) Modules. In order to communicate/access the MySQL databases, we’ll use mysql.connector module. MySQL-connector is a self-contained Python driver for communicating with MySQL servers to develop database applications. It is a DB API module that enables Python programs to access MySQL databases. It consists of different functions to perform various database operations.This module does not come built-in with Python.To install it, type the below command in the terminal/command prompt :-

#pip install mysql-connector

#Steps for connecting to MySQL database from within a Python Program:

Pre-requisites: mysql-connector must be installed in your machine.

1. import

2. create connection

#using connect()returns connection object

3. create cursor object

#using cursor()-returns cursor object

4. execute the query

#using execute()-accepts query as string literal

5. fetch result(s), if used select command

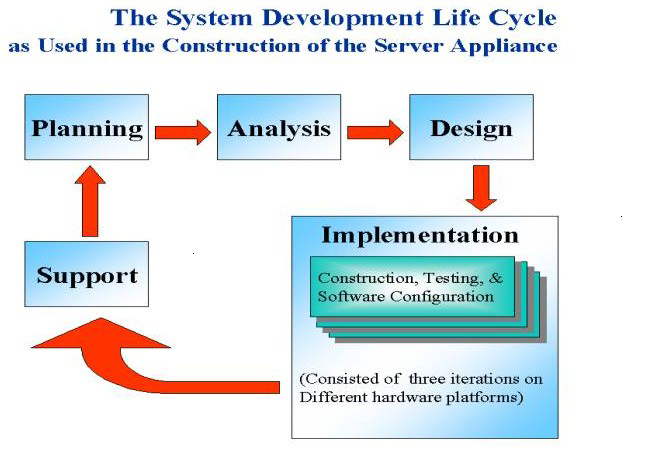
#fetchall(), fetchone(), fetchmany(n)

6. close the connection #using close()

**About RDBMS Backend (MySQL)**

Structured Query Language (SQL) is the most widely used commercial database language for creating, processing / manipulating and querying the relational DBMSs. Most of the RDBMS’s rely on SQL for their data processing capabilities. It was originally developed at IBM’s San Jose Research Laboratory in early 1970’s. This language was originally named as Sequel. The sequel language has evolved since then, and its name has been changed to SQL. SQL continues to evolve in response to the changing needs in the database area.

**SYSTEM DEVELOPMENT LIFE CYCLE (SDLC)**

****

The systems development life cycle is a project management technique that divides complex projects into smaller, more easily managed segments or phases. Segmenting projects allows managers to verify the successful completion of project phases before allocating resources to subsequent phases.

Software development projects typically include initiation, planning, design, development, testing, implementation, and maintenance phases. However, the phases may be divided differently depending on the organization involved.

For example, initial project activities might be designated as request, requirements-definition, and planning phases, or initiation, concept-development, and planning phases.

**PHASES OF SYSTEM DEVELOPMENT LIFE CYCLE**

**INITIATION PHASE**

The Initiation Phase begins when a business sponsor identifies a need or an opportunity.

The purpose of the Initiation Phase is to:

* Identify and validate an opportunity to improve business accomplishments of the organization or a deficiency related to a business need.
* Identify significant assumptions and constraints on solutions to that need.
* Recommend the exploration of alternative concepts and methods to satisfy the need including questioning the need for technology, i.e., will a change in the business process offer a solution?
* Assure executive business and executive technical sponsorship. The Sponsor designates a Project Manager and the business need is documented in a Concept Proposal. The Concept Proposal includes information about the business process andthe relationship to the Agency/Organization.
* Infrastructure and the Strategic Plan. A successful Concept Proposal results in a Project Management Charter which outlines the authority of the project manager to begin

the project.

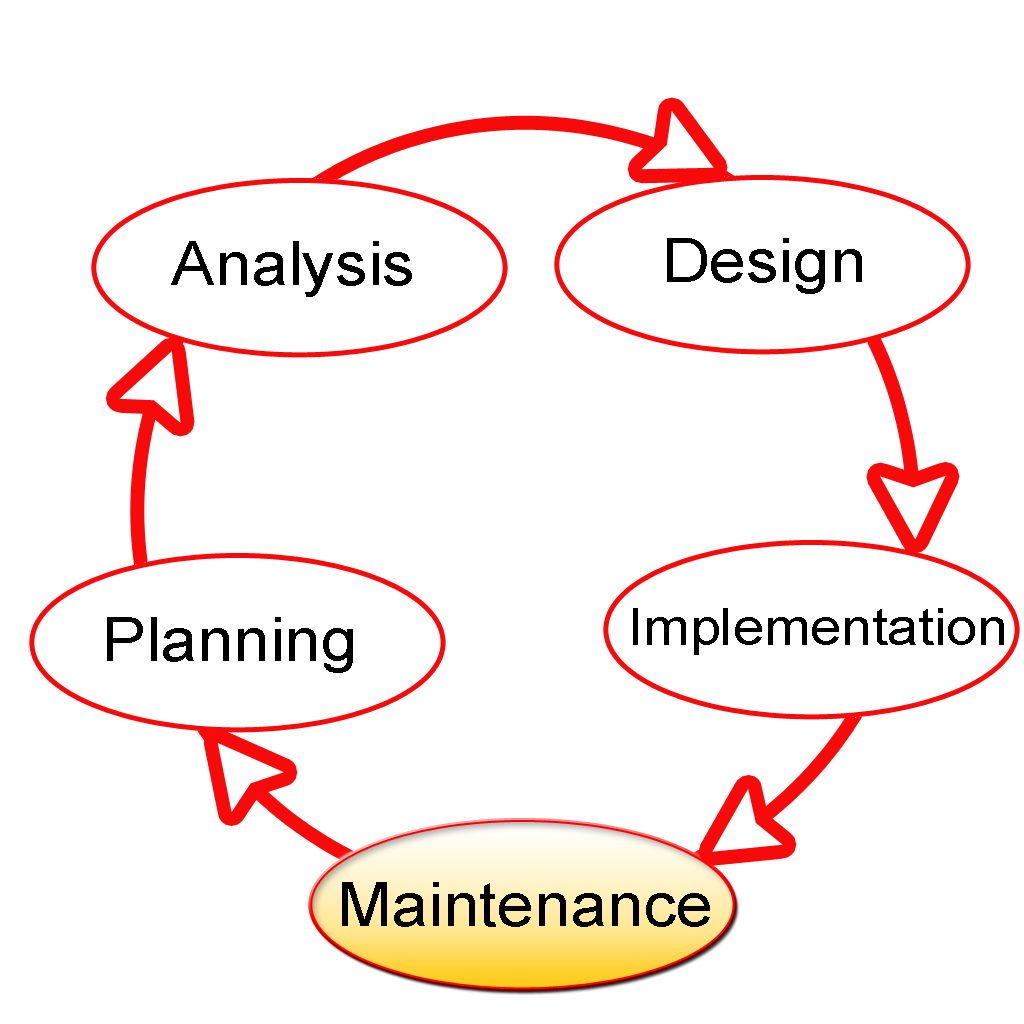
**SYSTEM CONCEPT DEVELOPMENT PHASE**

The System Concept Development Phase begins after a business need or opportunity is validated by the Agency/Organization Program Leadership and the Agency/Organization CIO.

The purpose of the System Concept Development Phase is to:

* Determine the feasibility and appropriateness of the alternatives.
* Identify system interfaces.
* Identify basic functional and data requirements to satisfy the business need.
* Establish system boundaries; identify goals, objectives, critical success factors, and performance measures.
* Evaluate costs and benefits of alternative approaches to satisfy the basic functional requirements
* Assess project risks
* Identify and initiate risk mitigation actions, andDevelop high-level technical architecture, process models, data models, and a concept of operations. This phase explores potential technical solutions within the context of the business need.

**PICTORIAL REPRESENTATION OF SDLC:**



**PLANNING PHASE**

The planning phase is the most critical step in completing development, acquisition, and maintenance projects. Careful planning, particularly in the early stages of a project, is necessary to coordinate activities and manage project risks effectively. The depth and formality of project plans should be commensurate with the characteristics and risks of a given project.

Project plans refine the information gathered during the initiation phase by further identifying the specific activities and resources required to complete a project.

**REQUIREMENTS ANALYSIS PHASE**

This phase formally defines the detailed functional user requirements using high-level requirements identified in the Initiation, System Concept, and Planning phases. It also delineates the requirements in terms of data, system performance, security, and maintainability requirements for the system. The requirements are defined in this phase to alevel of detail sufficient for systems design to proceed. They need to be measurable, testable, and relate to the business need or opportunity identified in the Initiation Phase. The requirements that will be used to determine acceptance of the system are captured in the Test and Evaluation MasterPlan.

The purposes of this phase are to:

* Further define and refine the functional and data requirements and document them in the Requirements Document,
* Complete business process reengineering of the functions to be supported (i.e., verify what information drives the business process, what information is generated, who generates it, where does the information go, and who processes it),
* Develop detailed data and process models (system inputs, outputs, and the process.
* Develop the test and evaluation requirements that will be used to determine acceptable system performance**.**

**DESIGN PHASE**

The design phase involves converting the informational, functional, and network requirements identified during the initiation and planning phases into unified design specifications that developers use to scriptprograms during the development phase. Program designs are c onstructed in various ways. Using a top-down approach, designers first identify and link major program components and interfaces, then expand design layouts as they identify and link smaller subsystems and connections. Using a bottom-up approach, designers first identify and link minor program components and interfaces, then expand design layouts as they identify and link larger systems and connections. Contemporary design techniques often use prototyping tools that build mock-up designs of items such as application screens, database layouts, and system architectures. Since problems in the design phase could be very expensive to solve in the later stage of the software development, a variety of elements are considered in the design to mitigate risk. These include:

* Identifying potential risks and defining mitigating design features.
* Performing a security risk assessment.
* Developing a conversion plan to migrate current data to the new system.
* Determining the operating environment.
* Defining major subsystems and their inputs and outputs.
* Allocating processes to resources.
* Preparing detailed logic specifications for each software module. The result is a draft System Design Document which captures the preliminary design for the system.

**DEVELOPMENT PHASE**

The development phase involves converting design specifications into executable programs. Effective development standards include requirements that programmers and other project participants discuss design specifications before programming begins. The procedures help ensure programmers clearly understand program designs and functional requirements. Programmers use various techniques to develop computer programs. The large transaction oriented programs associated with financial institutions have traditionally been developed using procedural programming techniques. Procedural programming involves the line-by-line scripting of logical instructions that are combined to form a program.Effective completion of the previous stages is a key factor in the success of the Development phase. The Development phase consists of:

* Translating the detailed requirements and design into system components.
* Testing individual elements (units) for usability.
* Preparing for integration and testing of the IT system.

**INTEGRATION AND TEST PHASE**

Subsystem integration, system, security, and user acceptance testing is conducted during the integration and test phase. The user, with those responsible for quality assurance, validates that the functional requirements, as defined in the functional requirements document, are satisfied by the developed or modified system. OIT Security staff assess the system security and issue a security certification and accreditation prior to installation /implementation. Multiple levels of testing are performed, including:

* Testing at the development facility by the contractor and possibly supported by end users.
* Testing as a deployed system with end users working together with contract personnel.
* Operational testing by the end user alone performing all functions. Requirements are traced throughout testing,a final Independent Verification & Validation evaluation is performed and all documentation is reviewedand accepted prior to acceptance of the system.

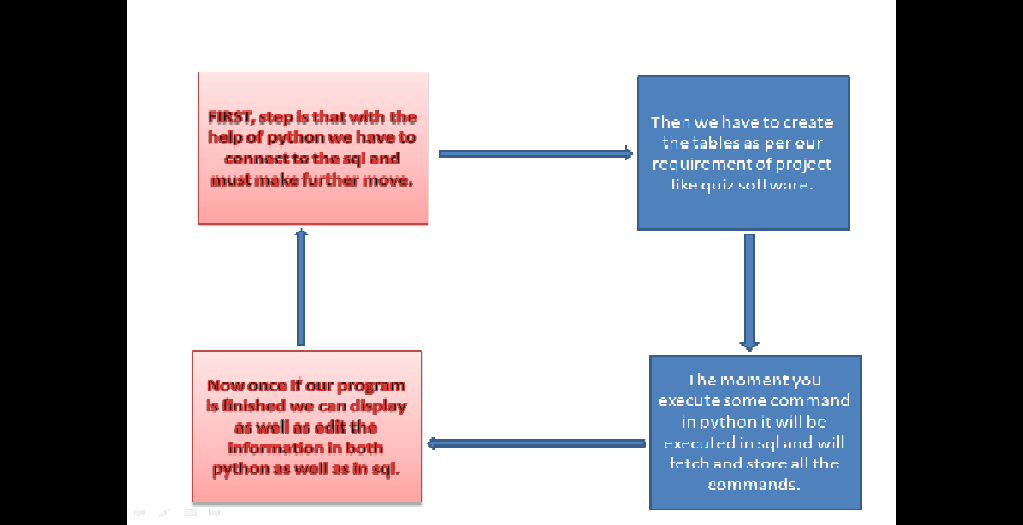
**IMPLEMENTATION PHASE**

This phase is initiated after the system has been tested and accepted by the user. In this phase, the system is installed to support the intended business functions. System performance is compared to performance objectives established during the planning phase. Implementation includes user notification, user training, installation of hardware, installation of software onto production computers, and integration of the system into daily work processes. This phase continues until the system is operating in production in accordance with the defined userrequirements.

**OPERATIONS AND MAINTENANCE PHASE**

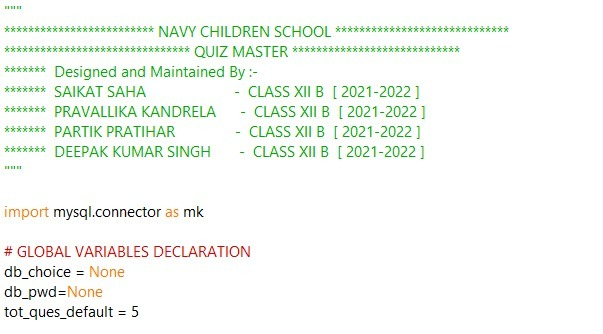
The system operation is ongoing. The system is monitored for continued performance in accordance with user requirements and needed system modifications are incorporated. Operations continue as long as the system can be effectively adapted to respond to the organization’s needs. When modifications or changes are identified, the system may reenter the planning phase. The purpose of this phase is to:

* Operate, maintain, and enhance the system.
* Certify that the system can process sensitive information.
* Conduct periodic assessments of the system to ensure the functional requirements continue to be satisfied.
* Determine when the system needs to be modernized, replaced, or retired.

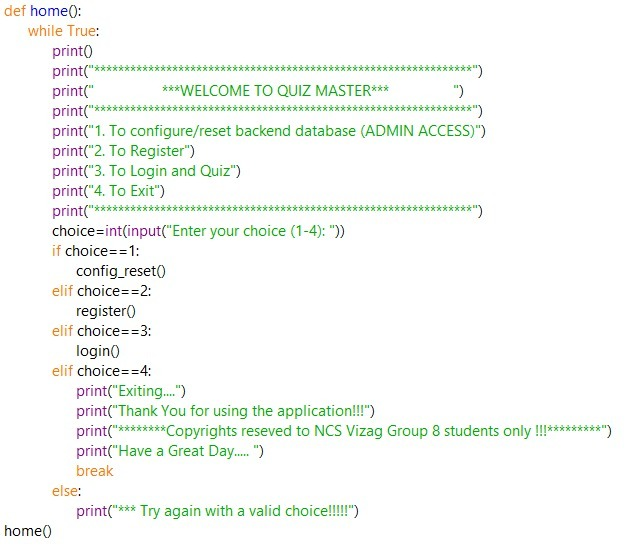
**FLOW CHART**

**SOURCE CODE**

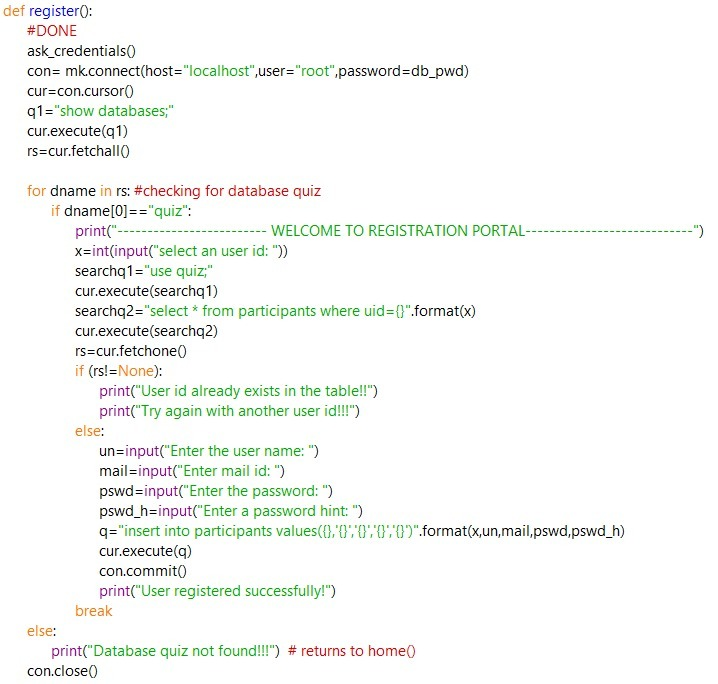
**### Description and Global variables**



**### MAIN MENU**



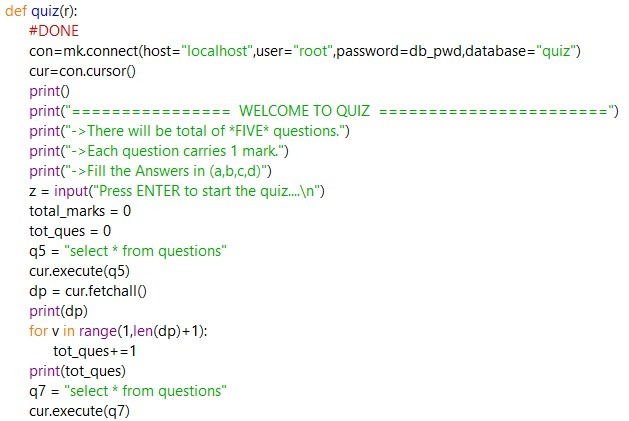
**## REGISTRATION**



**## LOGIN**

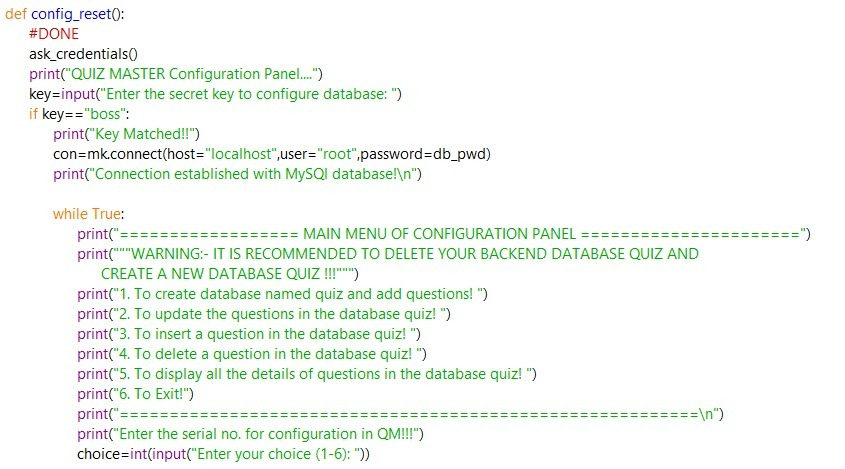


**### QUIZ**



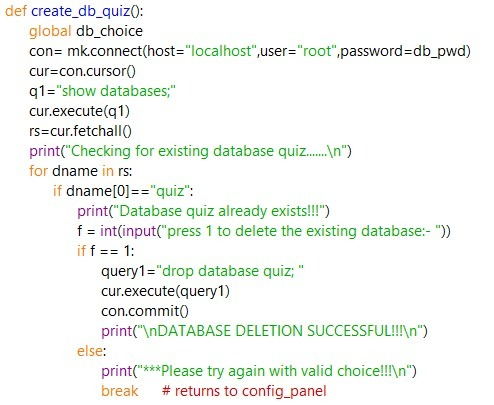


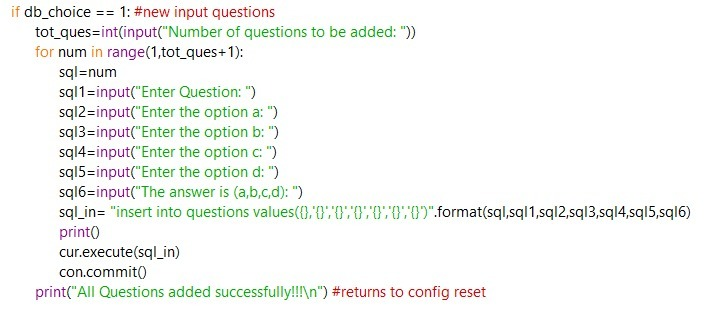
**## Quiz configuration panel**

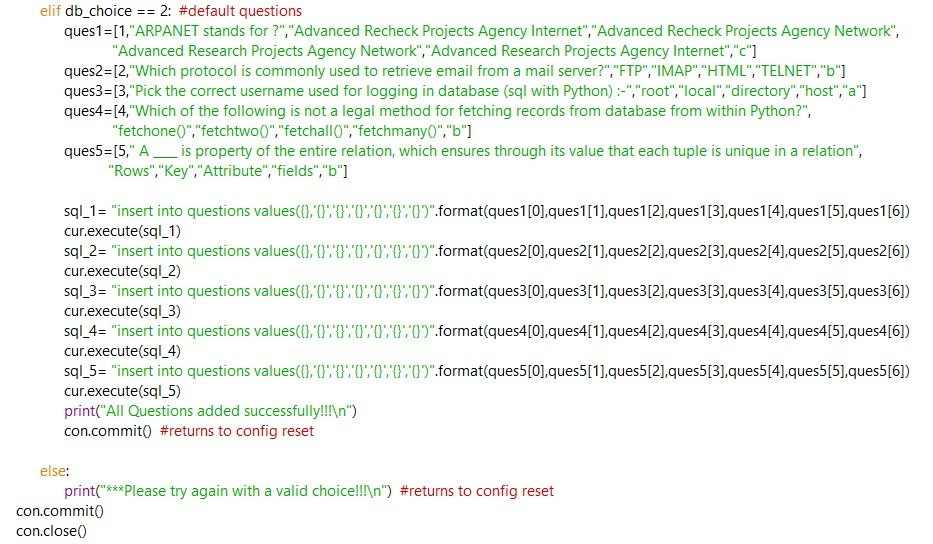


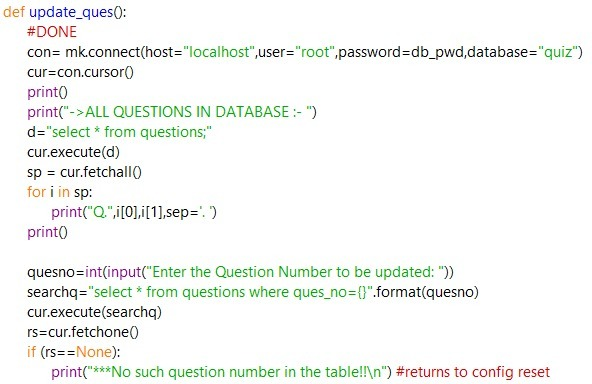


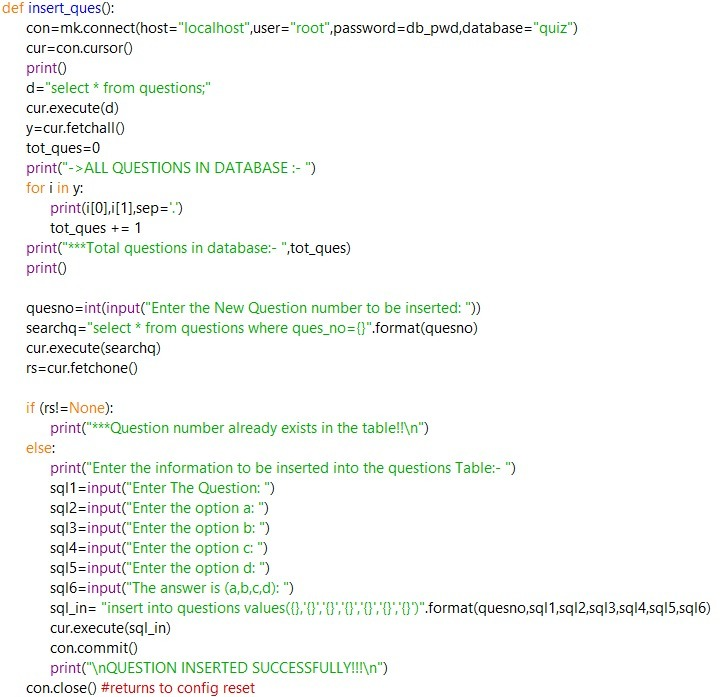
**# Admin configuration panel options**

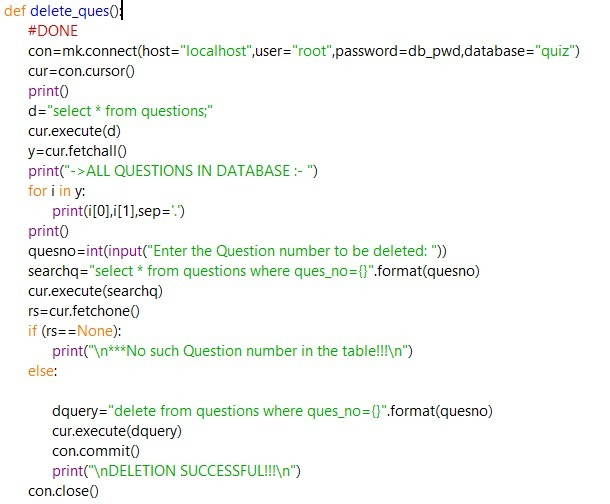


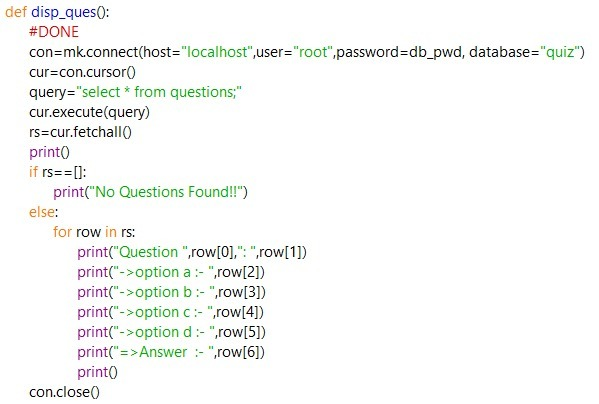






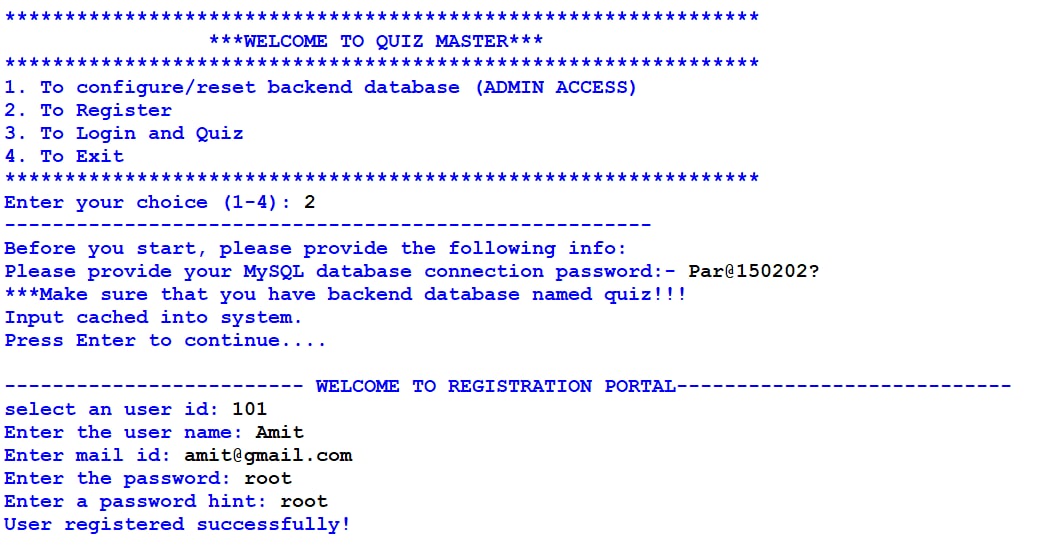




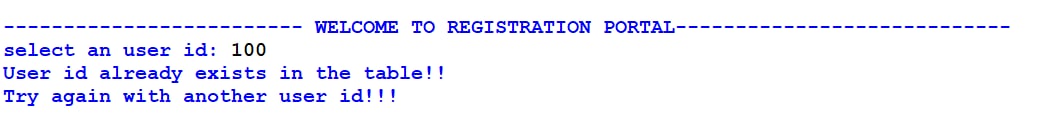


**OUTPUT**

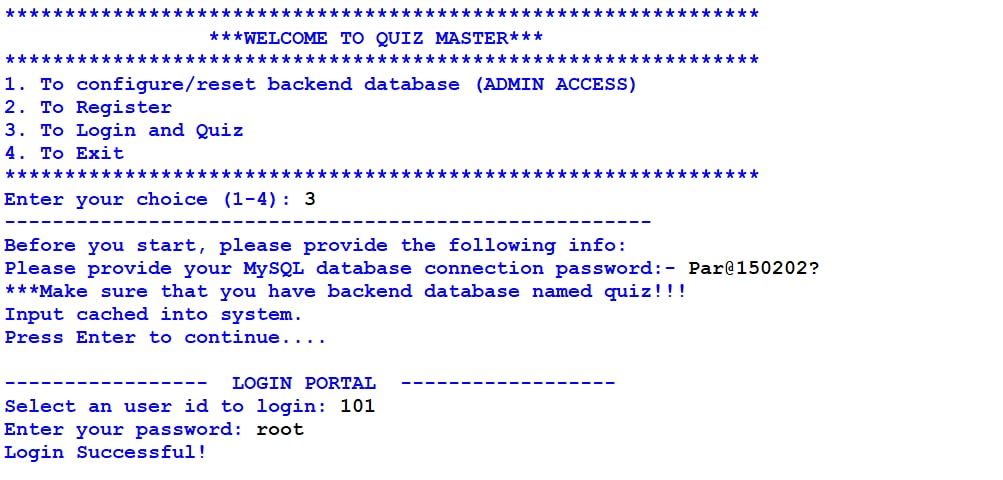
**### REGISTRATION PORTAL**

****

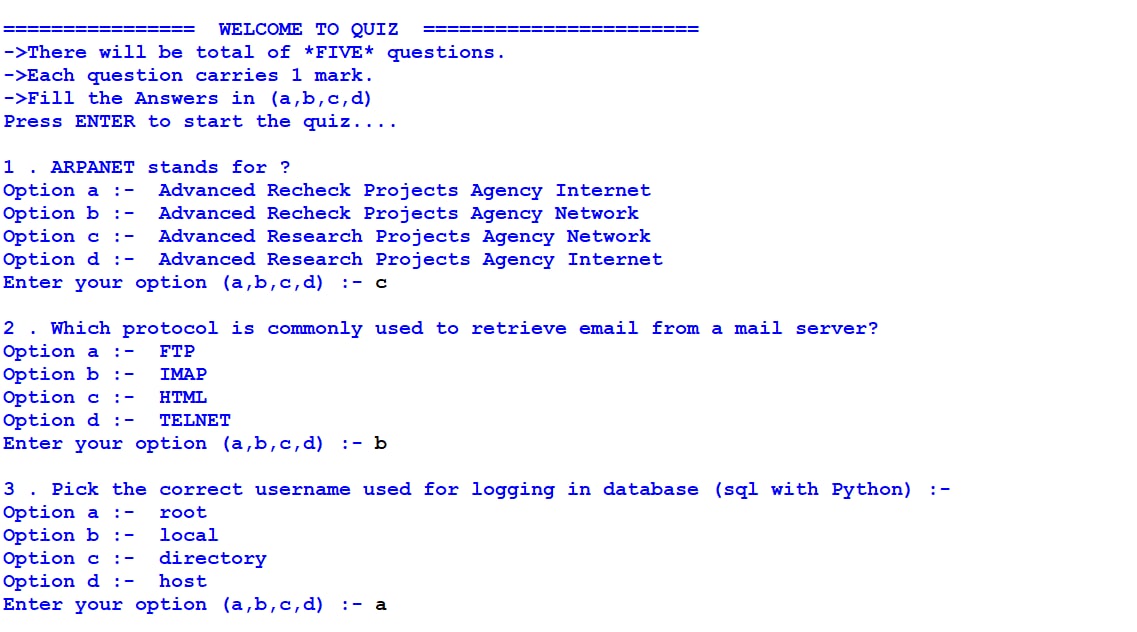
**# IF USER IS ALREADY REGISTERED**

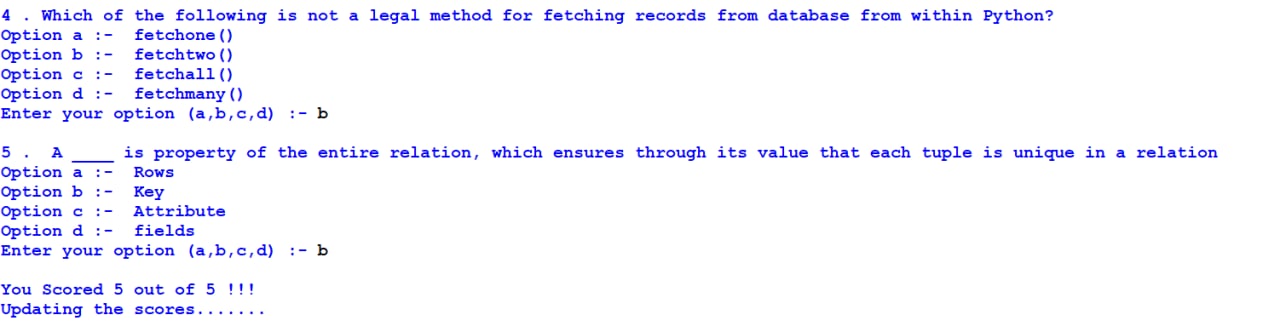
****

**### LOGIN PORTAL**

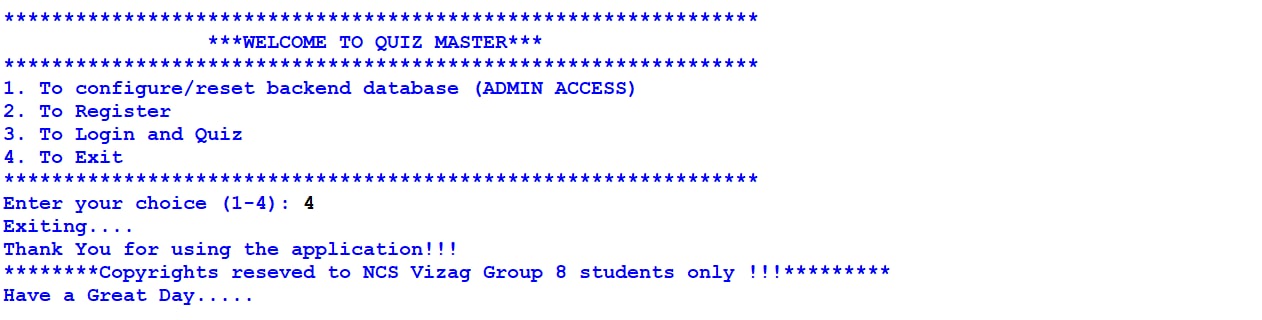
****

**### QUIZ PORTAL**

****

****

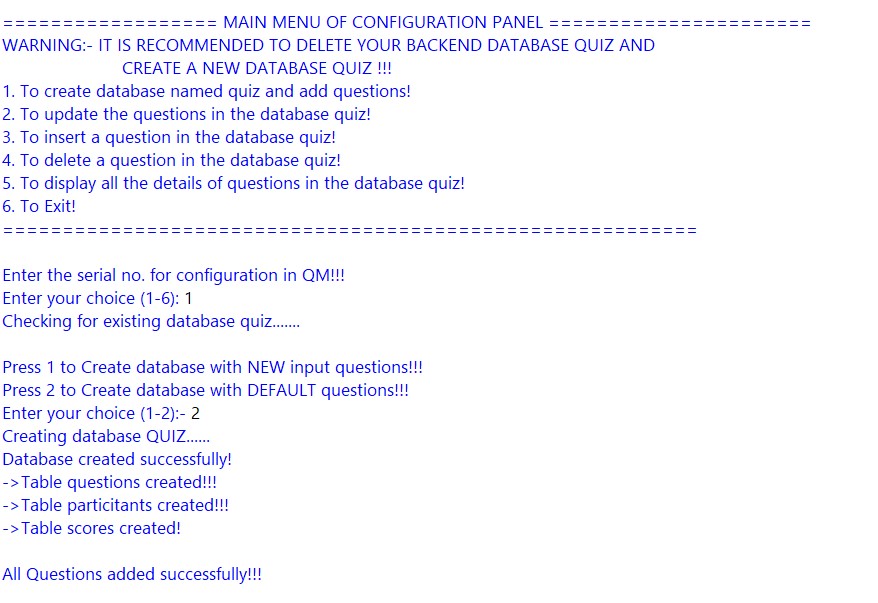
**### EXIT QUIZ MASTER**

****

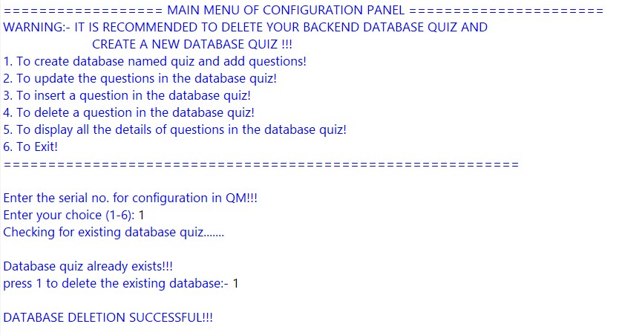
**### CONFIGURATION PANEL(ADMIN ACCESS)**

**# Creating database if database does not exist**

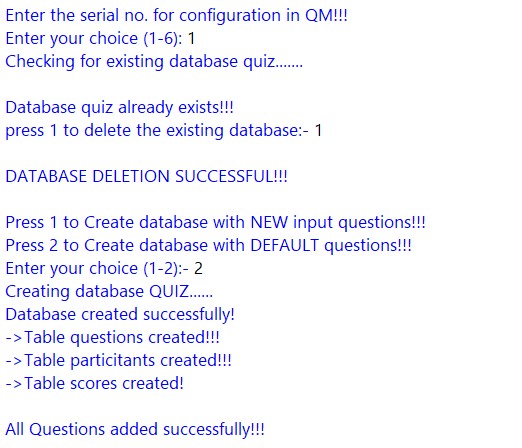
gfdg#########################



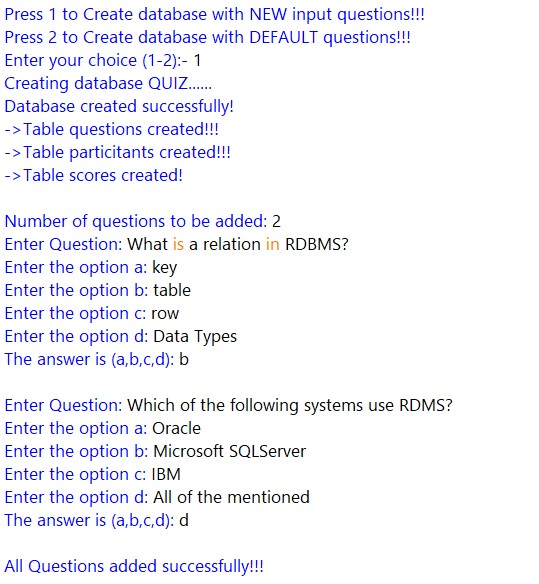
**# Creating new database by deleting the existing database**

****

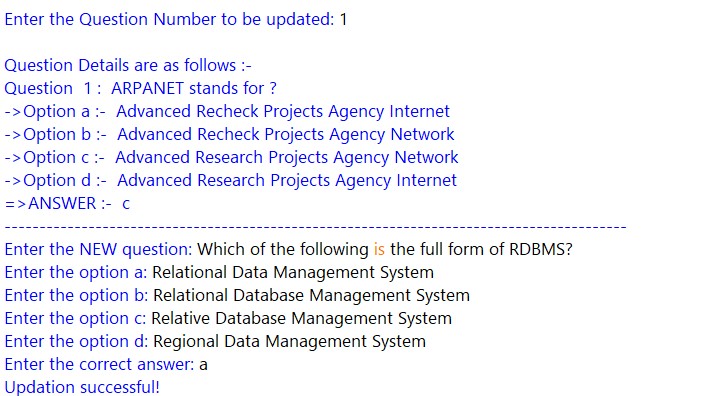
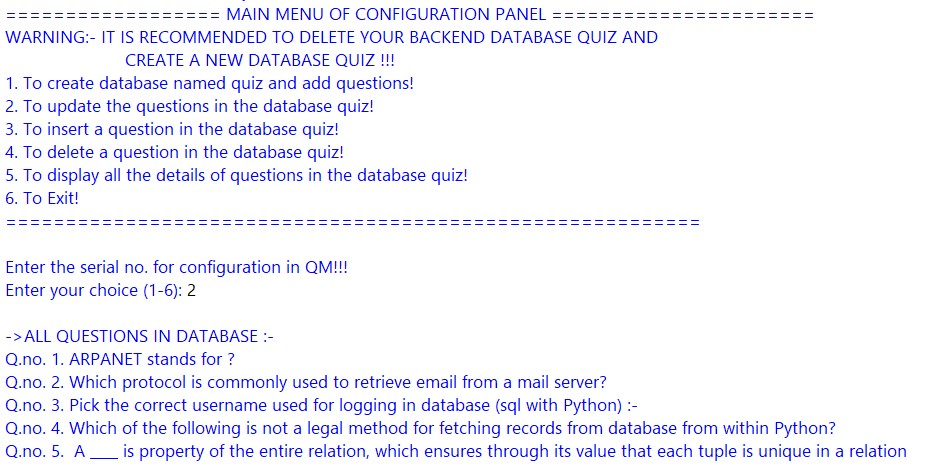
**#Creating database with default questions**



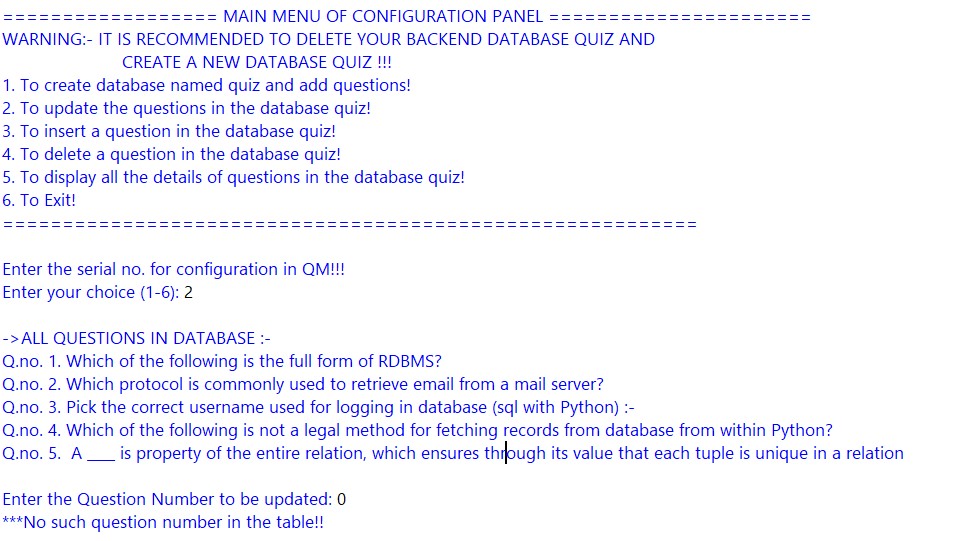
**#Creating database with admin questions**

****

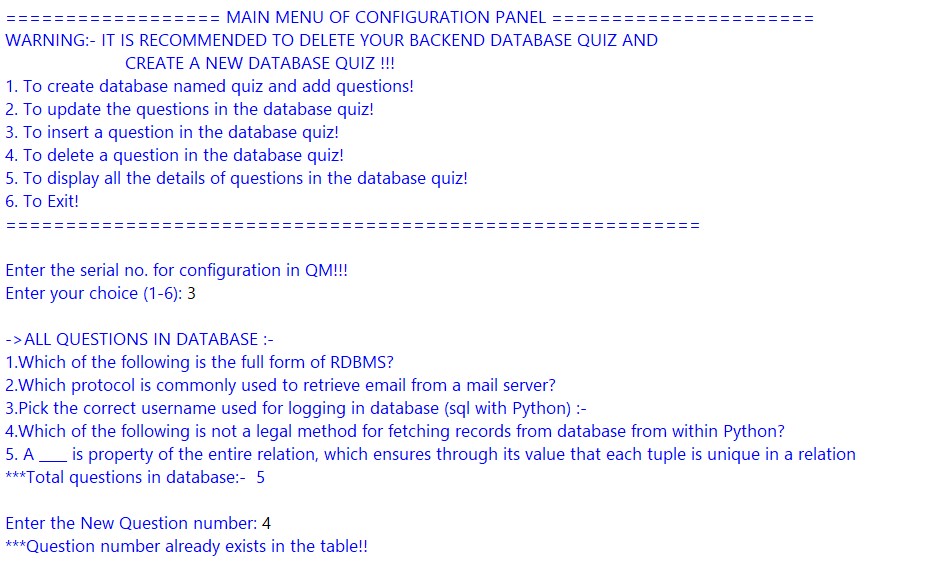
**#Updating the questions in database**



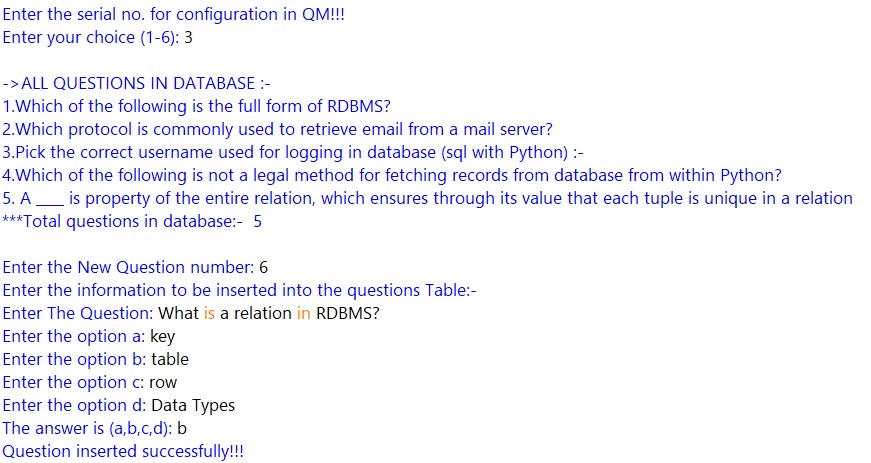
**# If question number to be updated does not exists in database**



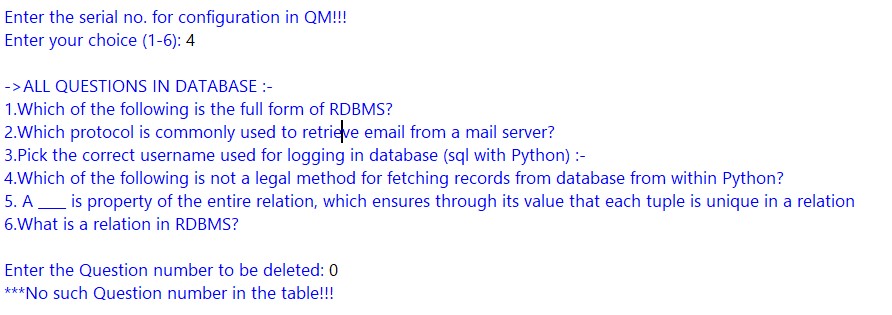
**# inserting existing question number in database**



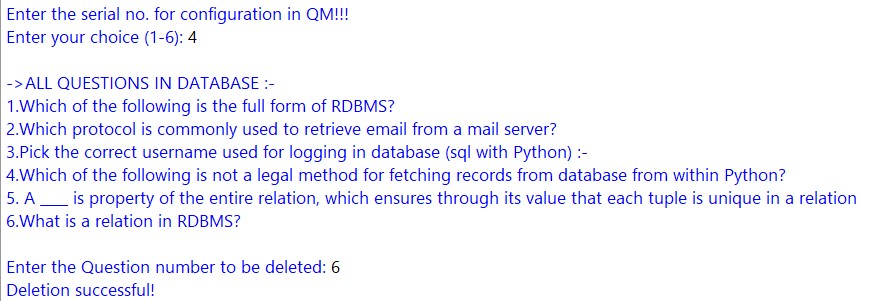
**#Inserting questions in database**



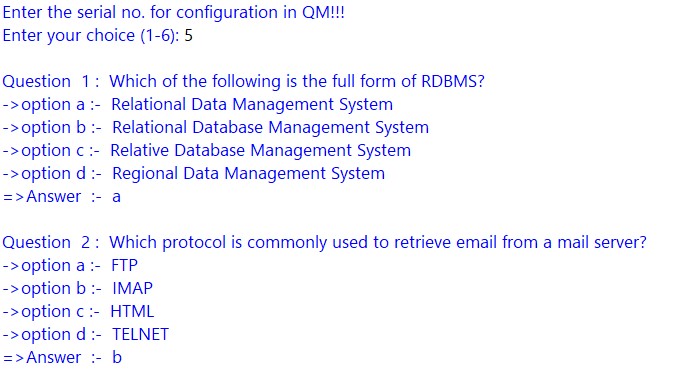
**#If the entered question number does not exists in database**

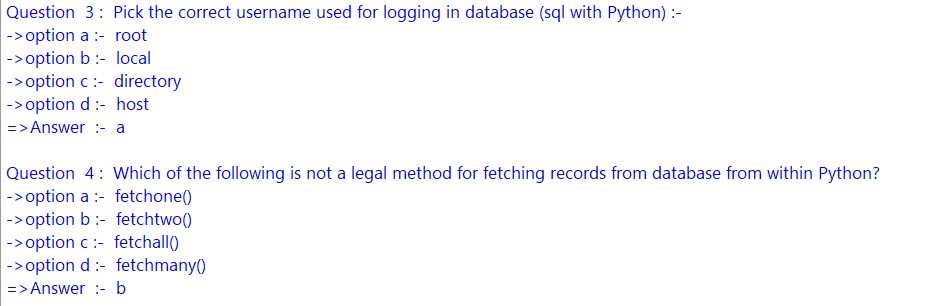


**#Deleting questions from database**

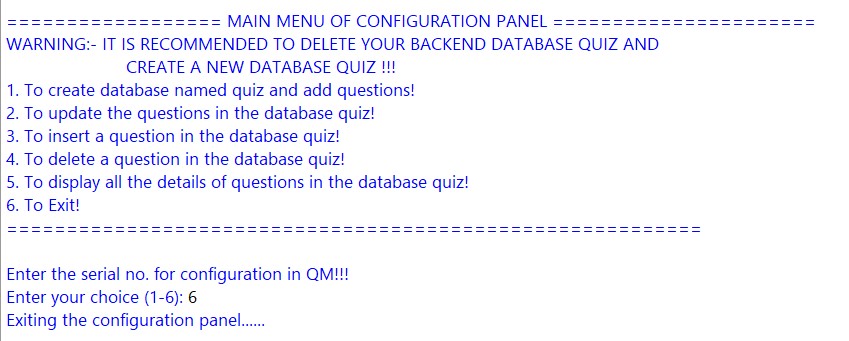


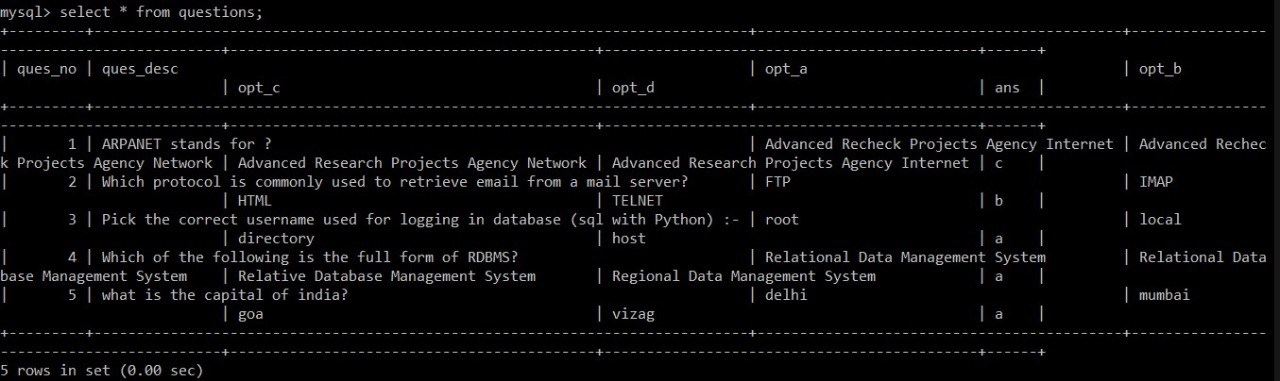
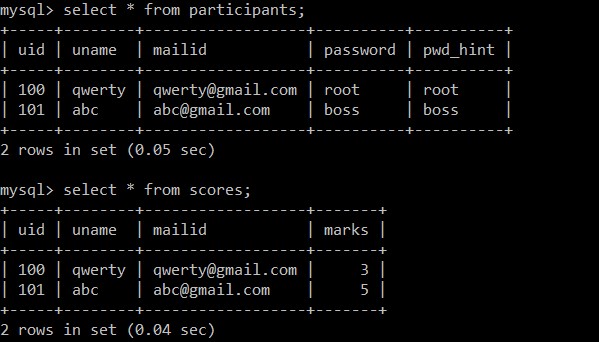
**#Displaying questions from database**

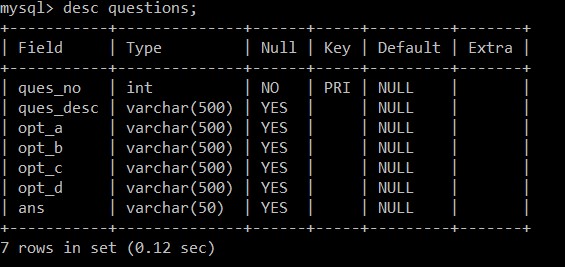
****

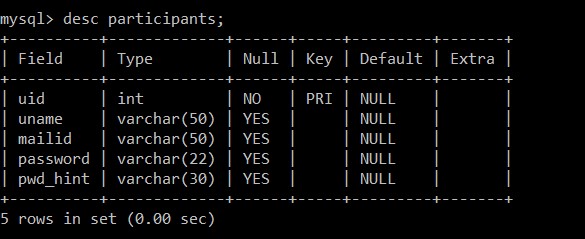


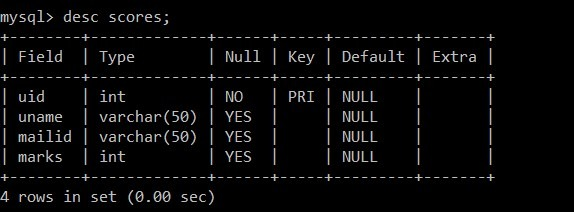
**#Exiting configuration panel**



**BLACK BOX OUTPUT**

****

****



**TESTING**

Software Testing is an empirical investigation conducted to provide stakeholders with information about the quality of the product or service under test[1] , with respect to the context in which it is intended to operate. Software Testing also provides an objective, independent view of the software to allow the business to appreciate and understand the risks at implementation of the software. Test techniques include, but are not limited to, the process of executing a program or application with the intent of finding software bugs.

It can also be stated as the process of validating and verifying that a software program/application/product meets the business and technical requirements that guided its design and development, so that it works as expected and can be implemented with the same characteristics. Software Testing, depending on the testing method employed, can be implemented at any time in the development process, however the most test effort is employed after the requirements have been defined and coding process has been completed.

**TESTING METHODS**

Software testing methods are traditionally divided into black box testing and white box testing. These two approaches are used to describe the point of view that a test engineer takes when designing test cases.

**BLACK BOX TESTING**

Black box testing treats the software as a "black box," without any knowledge of internal implementation. Black box testing methods include: equivalence partitioning, boundary value analysis, all-pairs testing, fuzz testing, model-based testing, traceability matrix, exploratory testing and specification-based testing.

**SPECIFICATION-BASED TESTING**

Specification-based testing aims to test the functionality of software according to the applicable requirements.[16] Thus, the tester inputs data into, and only sees the output from, the test object. This level of testing usually requires thorough test cases to be provided to the tester, who then can simply verify that for a given input, the output value (or behaviour), either "is" or "is not" the same as the expected value specified in the test case. Specification-based testing is necessary, but it is insufficient to guard against certain risks.

**ADVANTAGES AND DISADVANTAGES**

The black box tester has no "bonds" with the code, and a tester's perception is very simple: a code must have bugs. Using the principle, "Ask and you shall receive," black box testers find bugs where programmers don't. But, on the other hand, black box testing has been said to be "like a walk in a dark labyrinth without a flashlight," because the tester doesn't know how the software being tested was actually constructed.

That's why there are situations when (1) a black box tester writes many test cases to check something that can be tested by only one test case, and/or (2) some parts of the back end are not tested at all. Therefore, black box testing has the advantage of "an unaffiliated opinion," on the one hand, and the disadvantage of "blind exploring," on the other.

**WHITE BOX TESTING**

White box testing, by contrast to black box testing, is when the tester has access to the internal data structures and algorithms (and the code that implement these)

***Types of white box testing:-***

The following types of white box testing exist:

* api testing - Testing of the application using Public and Private APIs.
* Code coverage - creating tests to satisfy some criteria of code coverage.
* fault injection methods.
* mutation testing methods.
* static testing - White box testing includes all static testing.

**CODE COMPLETENESS EVALUATION**

White box testing methods can also be used to evaluate the completeness of a test suite that was created with black box testing methods. This allows the software team to examine parts of a system that are rarely tested and ensures that the most important function points have been tested. Two common forms of code coverage are:

* Function Coverage: Which reports on functions executed and
* Statement Coverage: Which reports on the number of lines executed to complete the test? They both return coverage metric, measured as a percentage.

**HARDWARE AND SOFTWARE**

**REQUIREMENTS**

**Hardware Requirements:**

|  |  |
| --- | --- |
| 1.Operating system | 1.Windows7 or above  2.Mac OS x 10.11 or above  3.Linux RHEL 6/7 |
| 2.Processor | 1.Intel Pentium or above  2.AMD Athlon or above  3.Nvidiategra or above |
| 3.Motherboard | 1.845 or 915,995 for Pentium or MSI  K9MM-V  VIAk8M8oo+8237P plus Chipsetfor AMB Athlon |
| 4.RAM | 512MB or above |
| 5.Hard disk | SATA 40 GB or above |
| 6.CD/DVD drive | If backup required |

**Software Requirements:**

1. Windows OS
2. Python
3. MySql

**BIBLIOGRAPHY**

1. **Computer science With Python - Class XII**

**By : Sumita Arora**

1. **Website :-**

**[www.w3resource.com](https://www.w3resource.com)**

**[www.google.com](http://www.google.com)**

**https://csstudy.in/**

***\*\*\* THE END \*\*\****