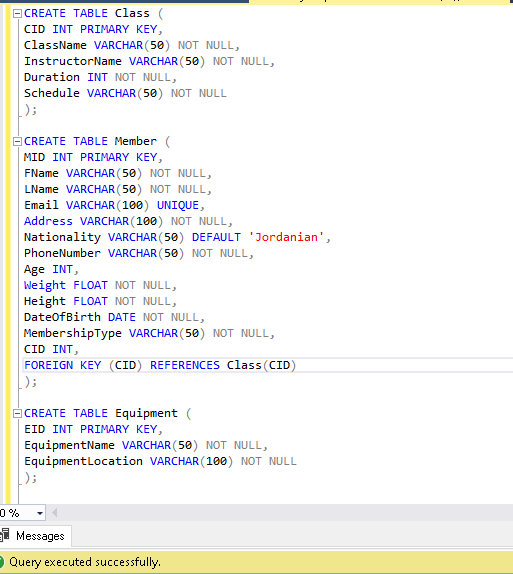
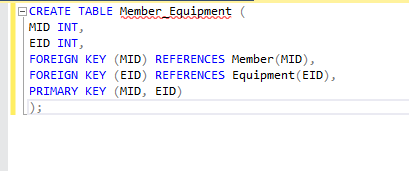
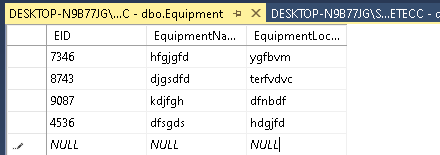
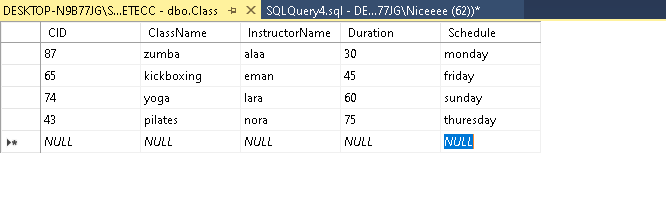
**Student Assessment Submission and Declaration**

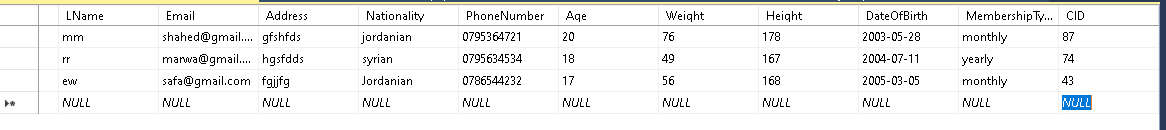
When submitting evidence for assessment, each student must sign a declaration confirming that the work is their own.

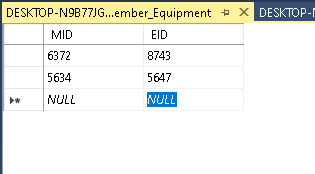
|  |  |  |  |
| --- | --- | --- | --- |
| Student name: Shahed Montaser | | Assessor name: **Hala Majdalawi** | |
| Issue date (1St Submission):  **22/5/2023** | Submission date (1St Submission):  **10/6/2023** | | Submitted on:  **10/6/2023** |
| In case of resubmission | | | |
| Issue date (1St Submission):  **13/6/2023** | Submission date (1St Submission):  **16/6/2023** | | Submitted on: |
| Programme: **HND in Cloud Computing - Cloud Software Development** | | | |
|  | | | |
| Assignment number and title: **2 Nitrogen Gym Cloud Database System** | | | |

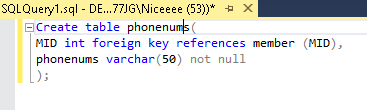
Task 2: Q1.

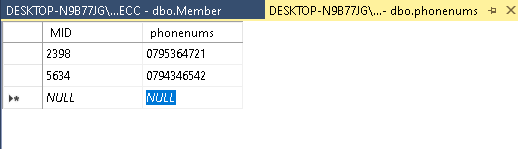




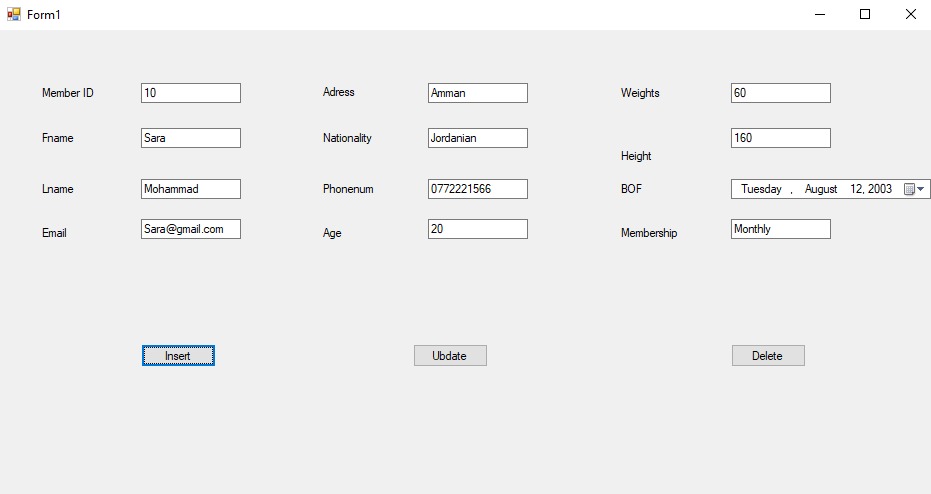


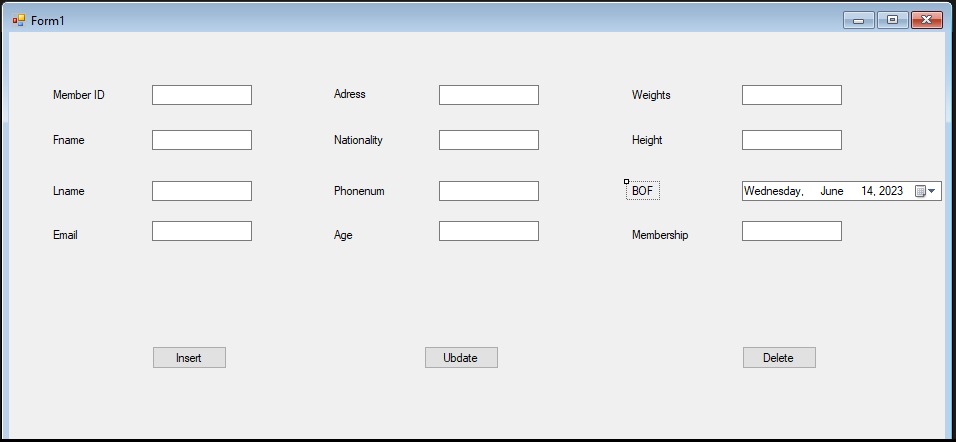




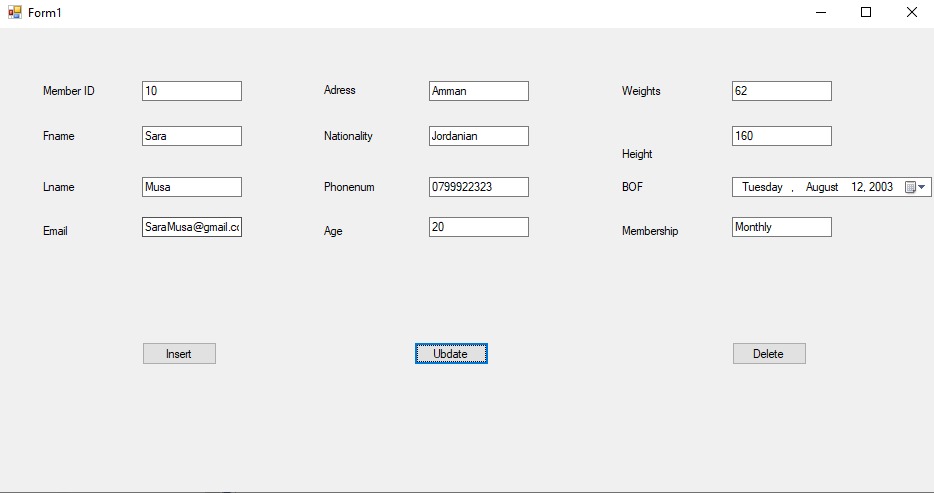


Q2.

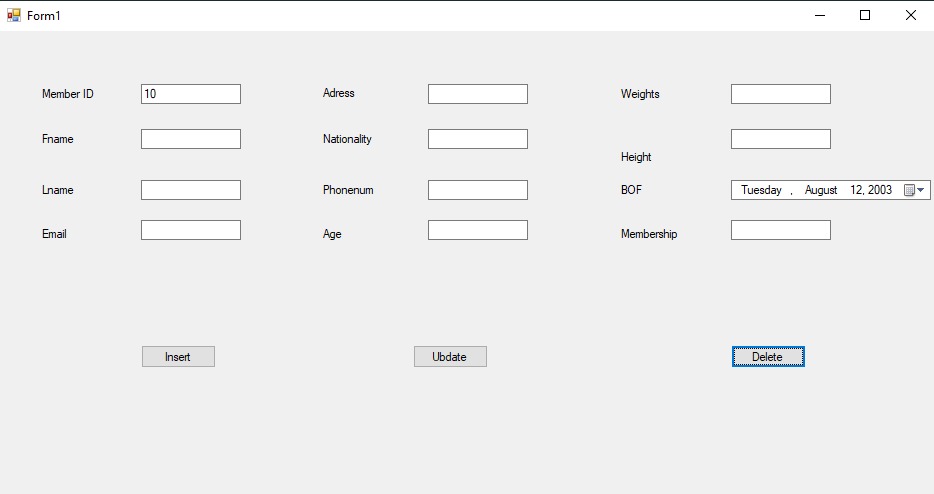




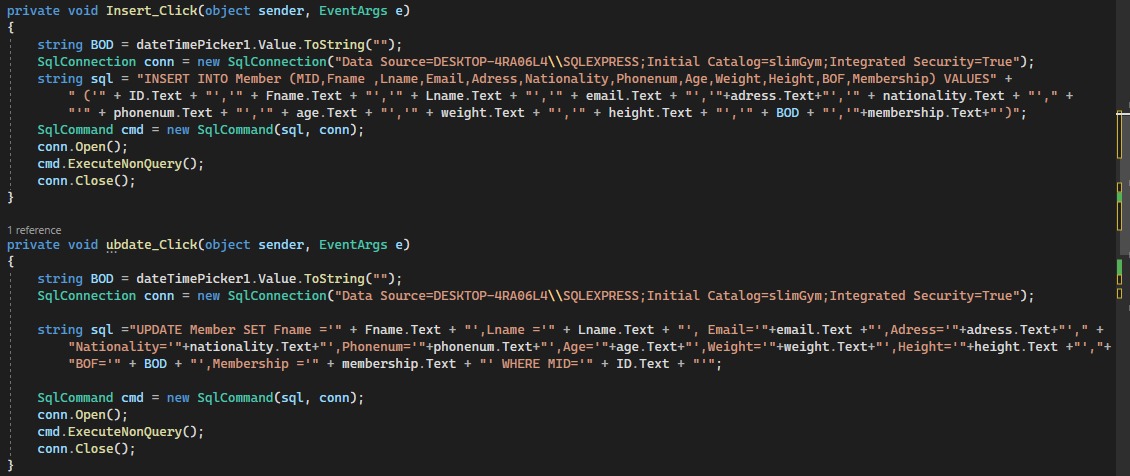










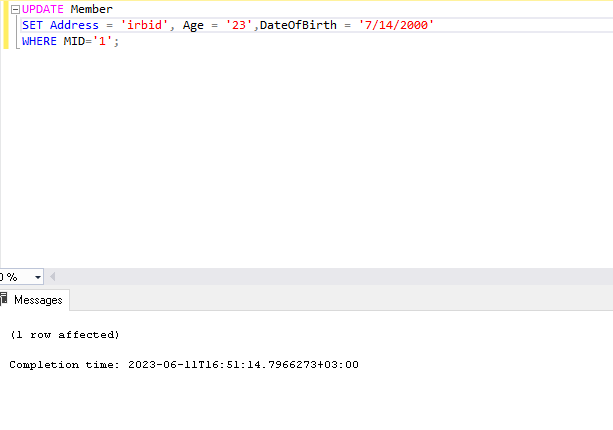


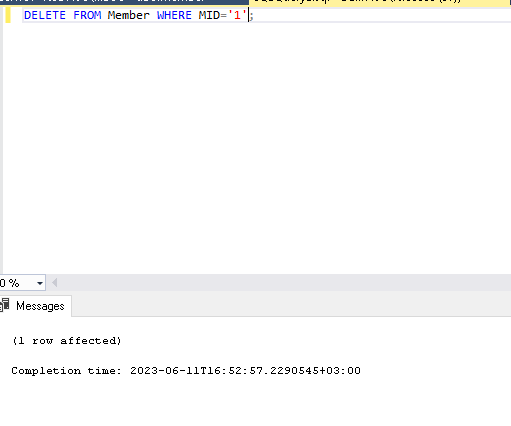


The previous screenshots from visual studio.

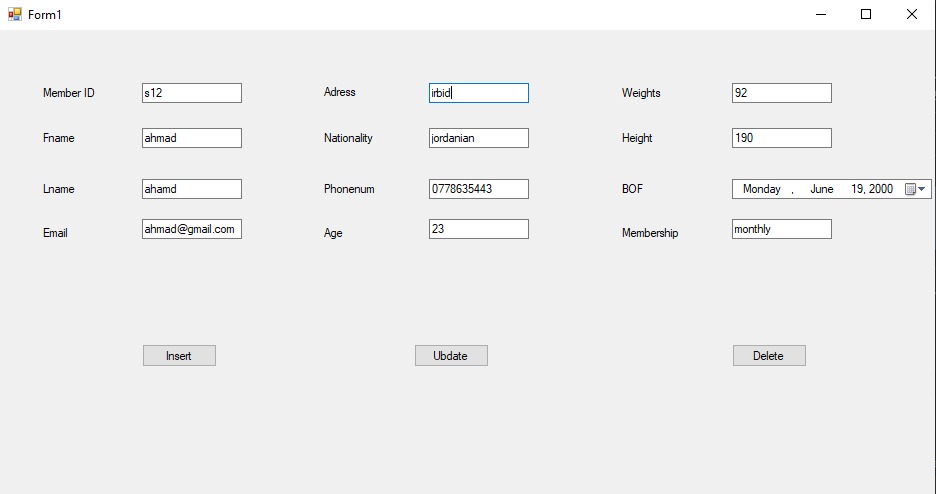
Now let’s see the process on SQL:



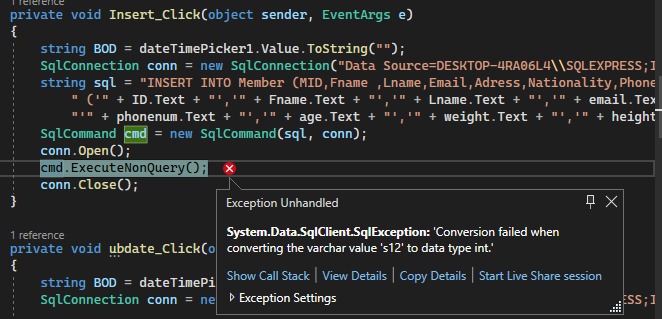




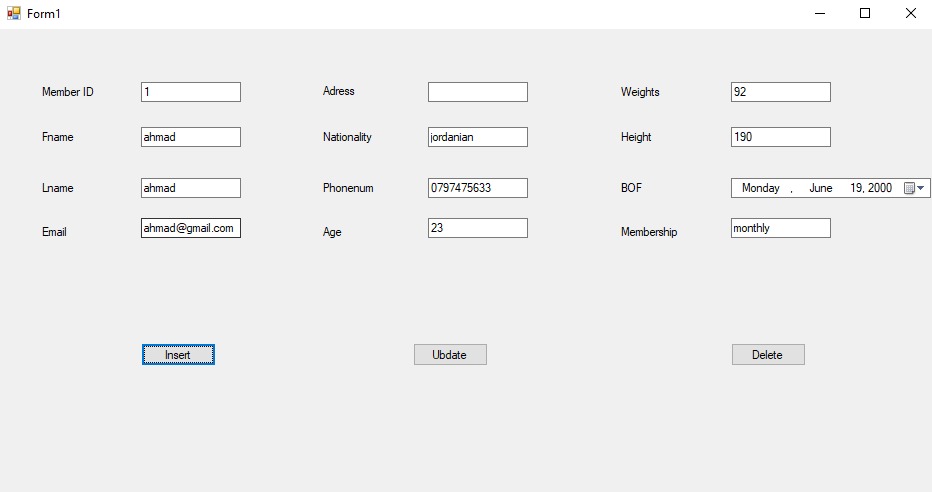
-implement the validation methods:



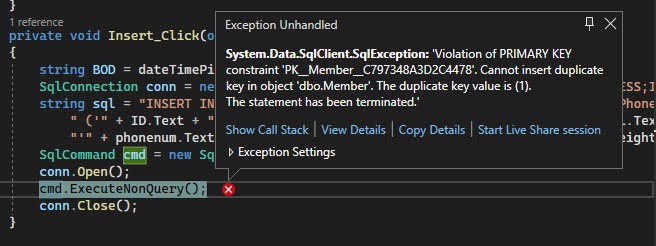
Here I entered a wrong ID containing letters to insert.



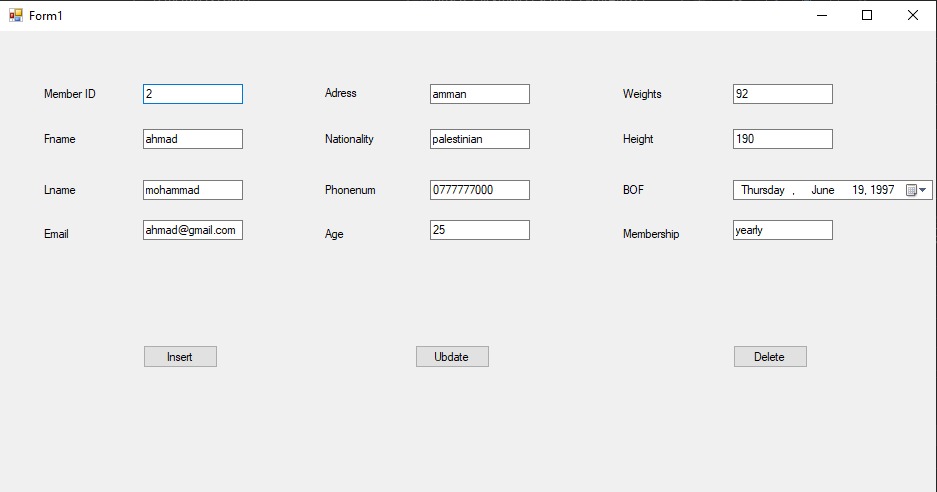
Here’s the error.



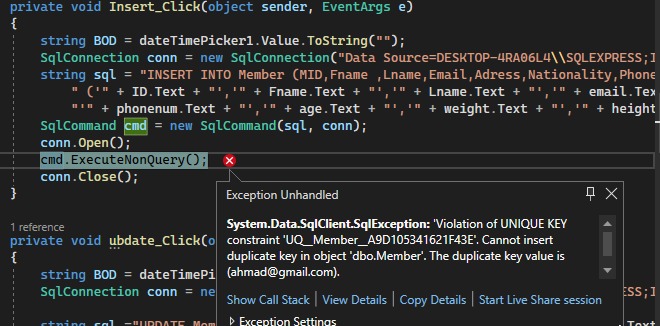
Here I entered a previously entered ID to insert(the primary key should be unique)



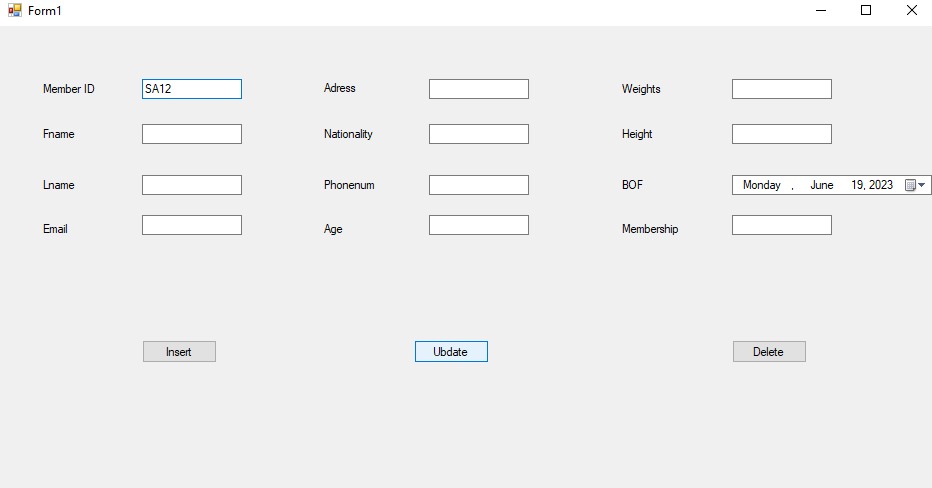
The error.



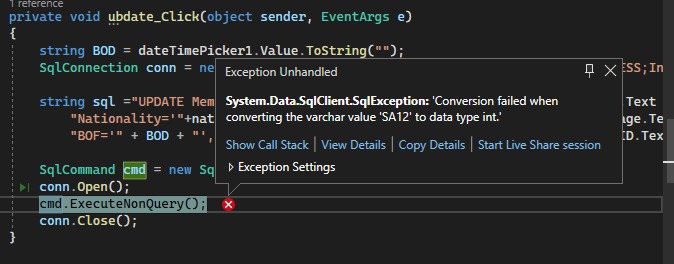
Here I entered a previously entered email for another person ( the e-mail must be unique).



Here’s the error.

ق

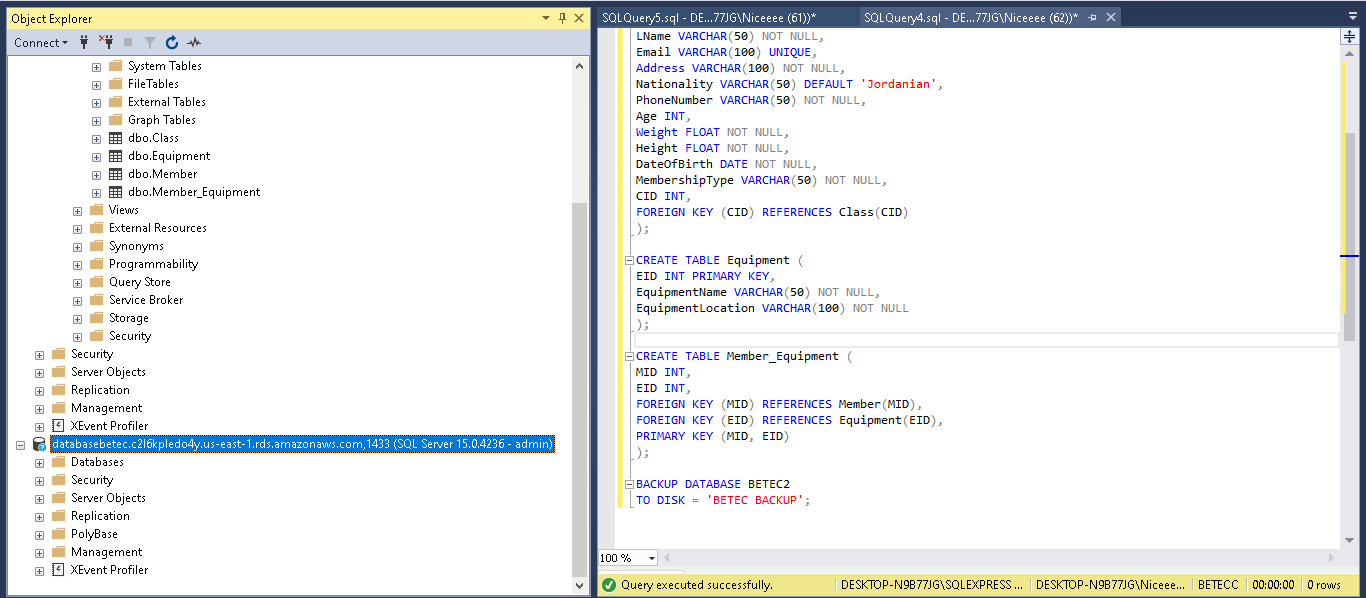
Here I entered an invalid ID to update.

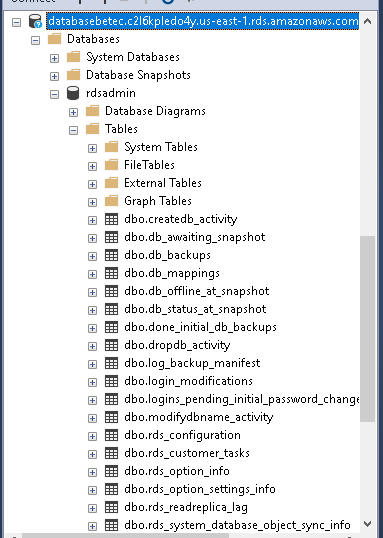


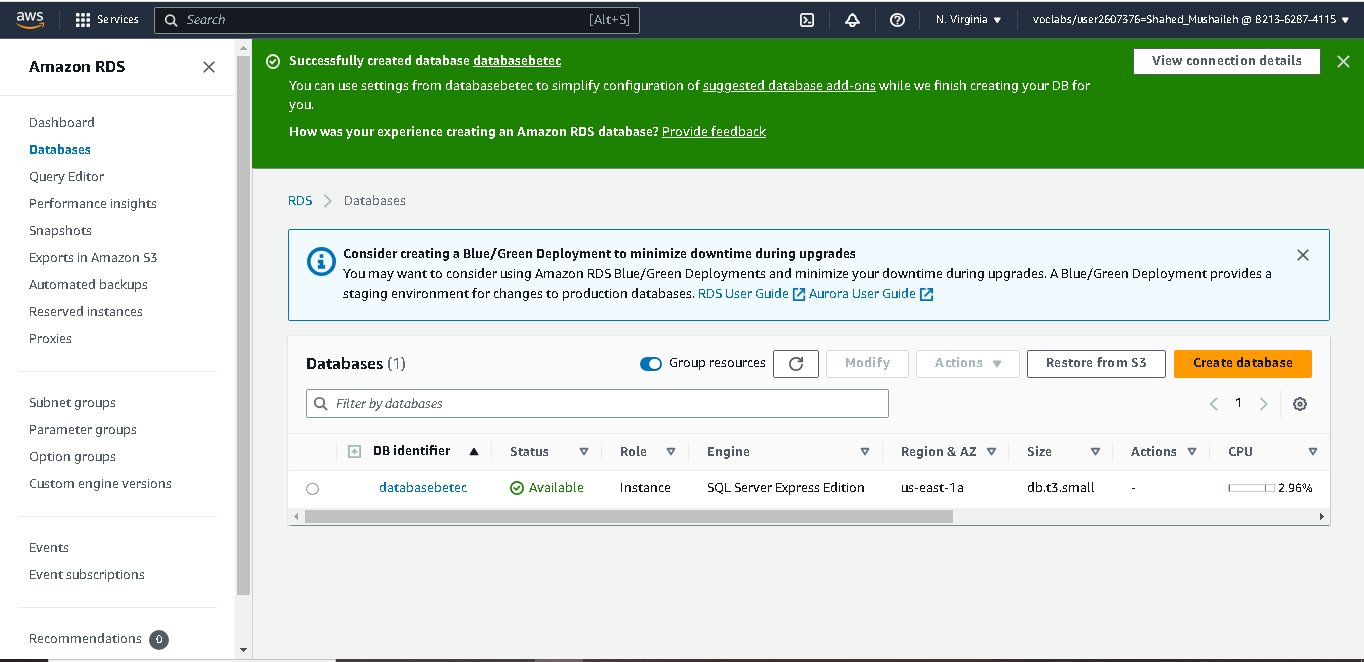
The error.

Q3.

Deploying the database on aws:





Q4.a.

Several security techniques may be used in the Nitrogen Gym cloud database system to guarantee the security and safety of the data. These controls are essential for protecting the system and preserving the privacy, accuracy, and accessibility of the data. Let's talk about some of the crucial security tools available:

1- Authentication: Before allowing users access to the system, authentication procedures are used to confirm their identity. Username and password authentication, MFA, or biometric authentication can all be used to accomplish this. The system can prevent unwanted access and defend against identity theft and credential-based threats by implementing robust authentication mechanisms.

2- Encryption: For sensitive data kept in the database to be protected, encryption is necessary. It includes applying cryptographic techniques to transform data into an unintelligible format. Without the encryption key, even if unauthorized parties obtain the material, they will not be able to decrypt it. This safeguards information privacy and stops unwanted dissemination.

3- Access Control: To limit and control access to the database system, access control techniques are used. Access control guarantees that only authorized people may carry out particular actions or access certain data by establishing user roles and privileges. Different roles may be allocated to users depending on their duties and permissions using role-based access control (RBAC). This makes sure that consumers, suppliers, and staff only have access to the information and features they need.

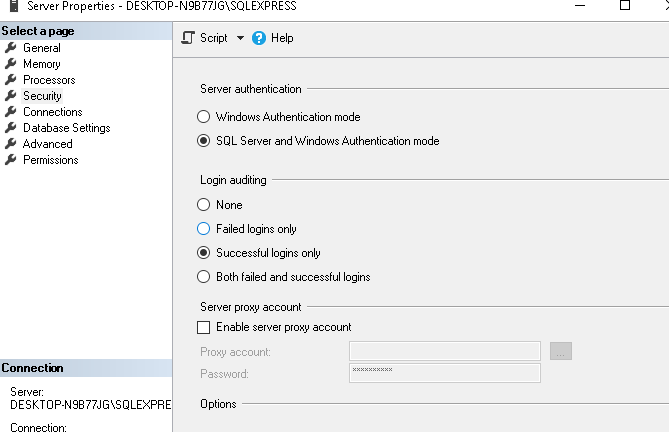
4- Backup and Disaster Recovery: For data protection and recovery in the case of system failures, data corruption, or security breaches, regular database backups are essential. Data restoration from a prior state is guaranteed through backup systems, limiting the impact of events and data loss. To manage major calamities and guarantee company continuity, off-site backups and disaster recovery strategies should be in place.

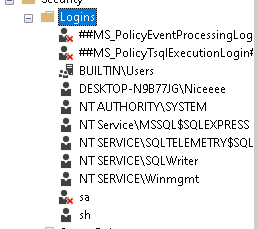
5- Regular upgrades and Patch Management: For resolving known vulnerabilities and lowering the risk of exploitation, it's essential to keep the database system updated with the most recent security patches and upgrades. Regular updates make sure that security flaws are closed and that the system is still able to fend off new dangers.

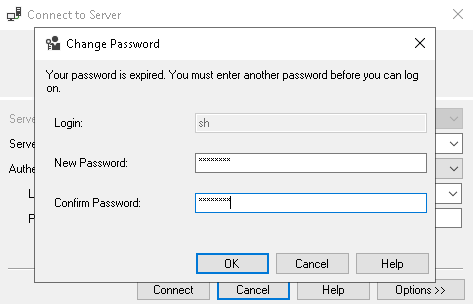
6- Auditing and Logging: Mechanisms for auditing and logging system actions, such as user logins, data alterations, and access attempts, are used to monitor and record these events. Any suspicious or harmful activity may be found and looked into by keeping thorough audit records. Additionally, logging aids forensic investigation and security rule compliance.

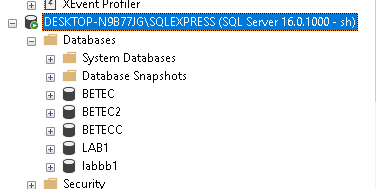
These security measures are crucial for safeguarding private member information, equipment details, and other vital elements of the Nitrogen Gym cloud database system. The system may avoid unwanted access, data breaches, information leaks, and significant losses in money or reputation by putting strong security measures in place. Additionally, adhering to security best practices and following applicable laws enhances the organization's reputation by instilling trust in members, workers, and stakeholders.

Q4.b.password

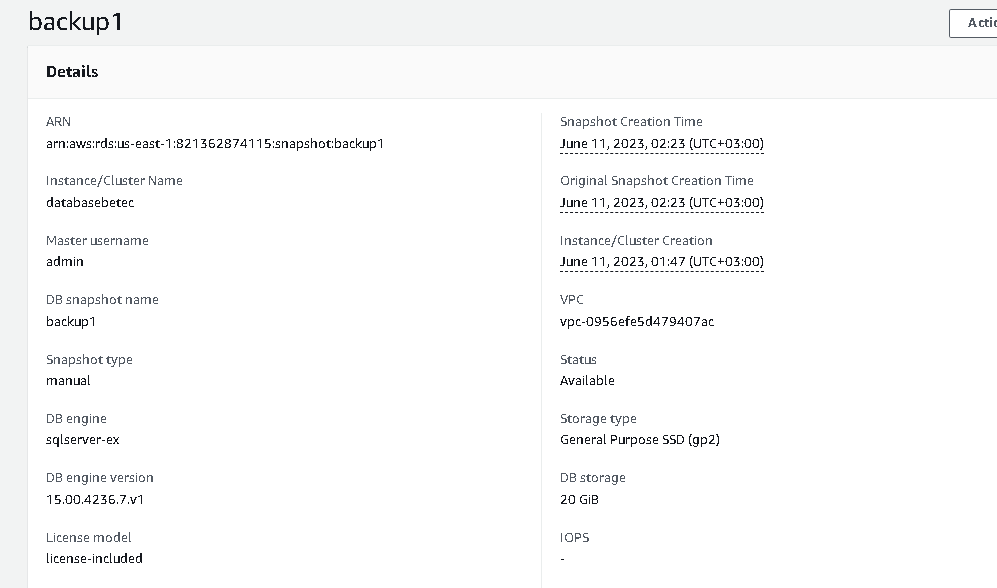


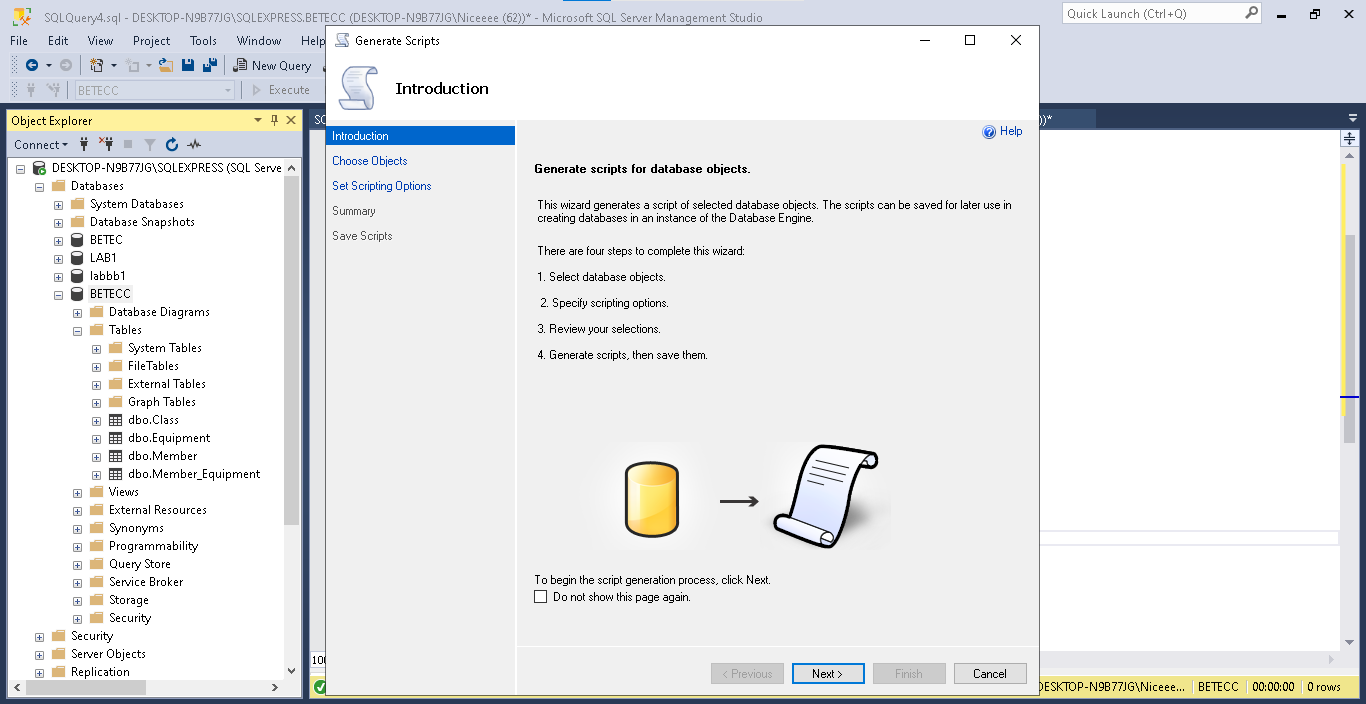


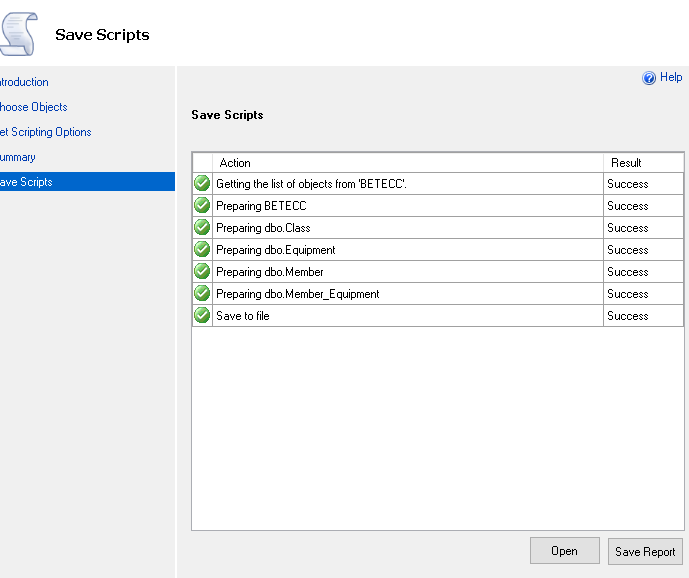


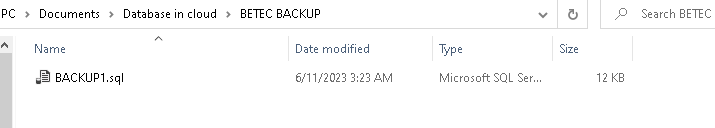


Q4.c. backup









Q.5.

A summary of how to assess the latency and accessibility of the performance of your cloud database system:

\*Latency Evaluation:

-A request's latency is the amount of time it takes to go from the client to the server and back. It is possible to pinpoint potential bottlenecks and improve response times by measuring latency.

-Utilize a tool that gives latency measures for performance monitoring or database monitoring. These instruments can monitor query execution times and offer perceptions of overall database performance.

-In order to spot any particular regions that could need optimization, keep an eye on the latency of various sorts of queries and processes, such as data retrieval, updates, and complicated joins.

-Set latency thresholds or warnings. If the delay is greater than certain criteria, there is a performance problem that has to be fixed.

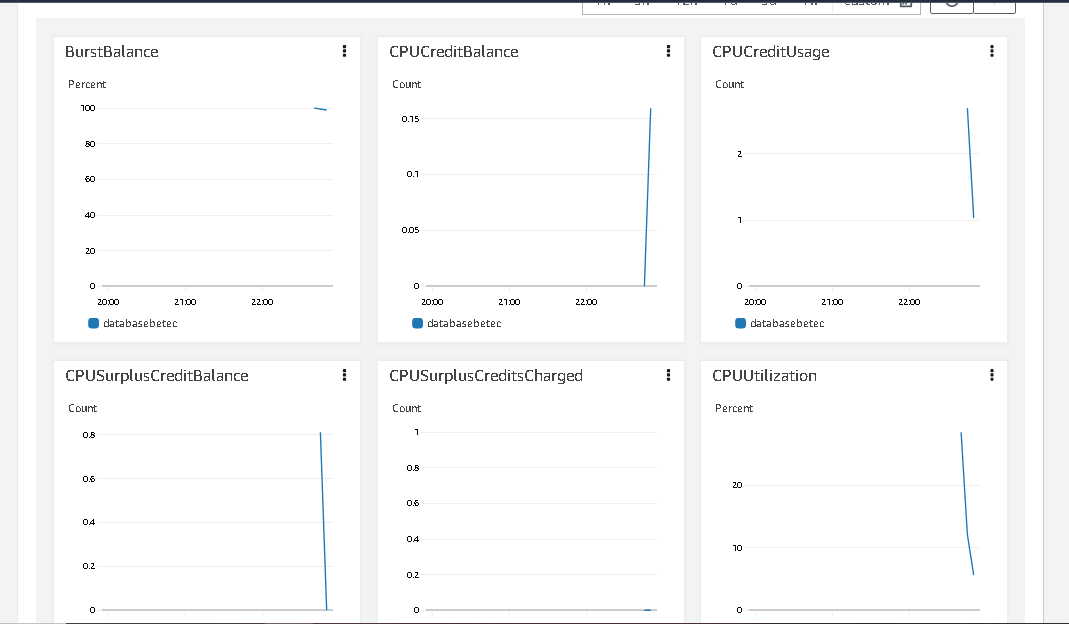
\*Availability Evaluation:

-Accessibility is the capacity to regularly and reliably access the cloud database system.

-Install a monitoring system that periodically examines the database server's accessibility. This may be achieved by regularly checking if the server is reachable using pings or requests.

-Set up automated notifications to inform you in the event of any outage or accessibility concerns. This makes rapid troubleshooting and reaction possible.

-To verify the system's capacity to manage the anticipated demand, simulate several concurrent users and heavy traffic situations using load testing tools. During these tests, record reaction times and mistakes to evaluate accessibility under various circumstances.



- We may take into account the metrics of BurstBalance, CPUCreditBalance, CPUCreditUsage, CPUUtilization, latency, and accessibility while assessing the performance of your Nitrogen Gym cloud database system. Here is a breakdown of each measure and how it relates to the effectiveness of the system:

\* BurstBalance:

Elastic Block Store (EBS) volumes on Amazon Web Services (AWS) use the BurstBalance measure.

It assesses your EBS volume's burst credit availability, which has an impact on its capacity to manage heavy workloads.

By keeping an eye on the BurstBalance, you can make sure that your EBS volume is performing well enough when there is a lot of I/O going on.

And the contrast between the images above and below is evident.

\*CPUCreditBalance:

The quantity of earned CPU credits that a given instance has accrued is represented by its CPUCreditBalance. These credits may be utilized to surpass the performance threshold.

\* CPUCreditUsage:

This metric shows how many CPU credits the instance has used up.

We can determine how CPU credits are being used and whether your instances are constantly using their allotted resources by keeping an eye on these indicators.

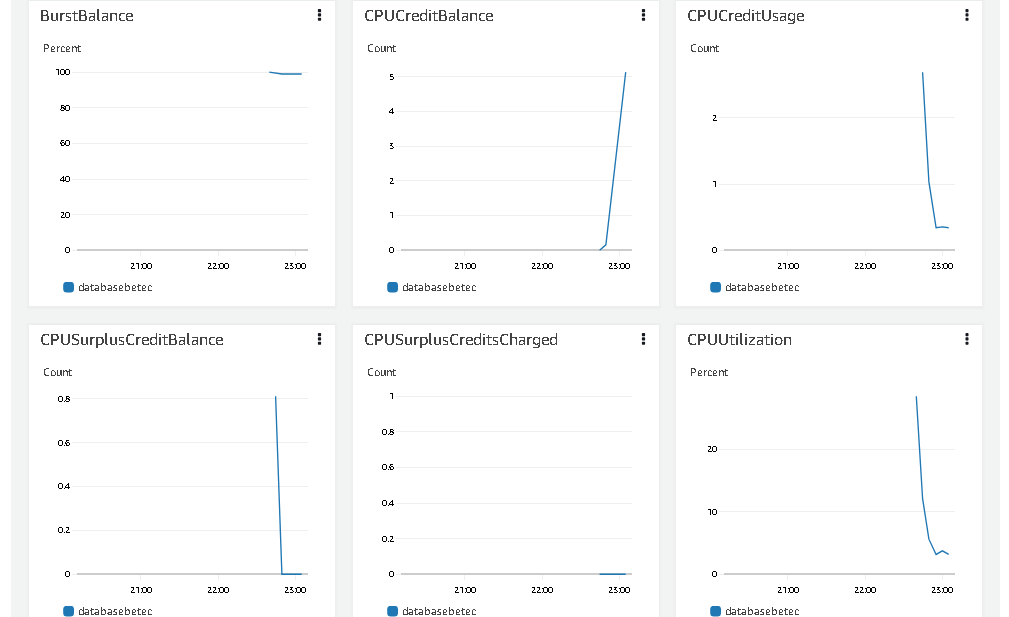
\* CPUUtilization:

The amount of CPU capacity that your instances are using is tracked by CPUUtilization.

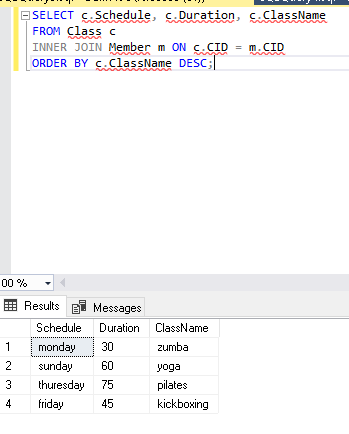
It displays the CPU use percentage and may be used to spot instances of excessive load or possible performance bottlenecks.

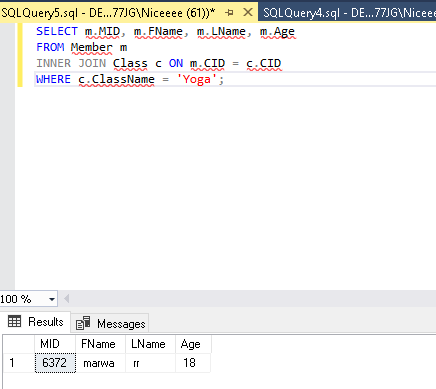
Making ensuring your instances have enough resources to tackle the job without being overloaded is made easier by keeping an eye on CPUUtilization.

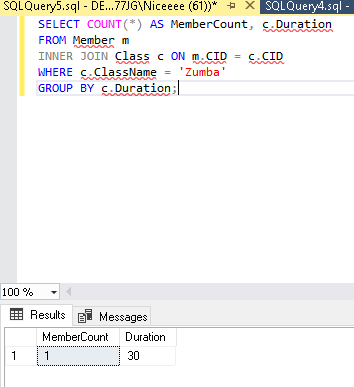
-By regularly monitoring and analyzing these metrics, you can identify performance issues, optimize resource allocation, and ensure the accessibility and responsiveness of your Nitrogen Gym cloud database system.



Q.6.







Q.7.

Testing:

To check and test the Nitrogen Gym cloud database system against the provided user and system requirements, we can perform the following checks:

1- Add and edit personal details:

I achieved this in a user interface made by visual studio.

2- Membership type selection:

I achieved this in a user interface made by the visual studio and in the design of the first task.

3- Lesson planning and reservation:

I achieved this in a user interface made by the visual studio and in the query of the database.

4- User-friendly interface.

5- Privacy settings and data security:

I applied a password for the database and deployed it on the cloud for more security.

6- Scalability and performance.

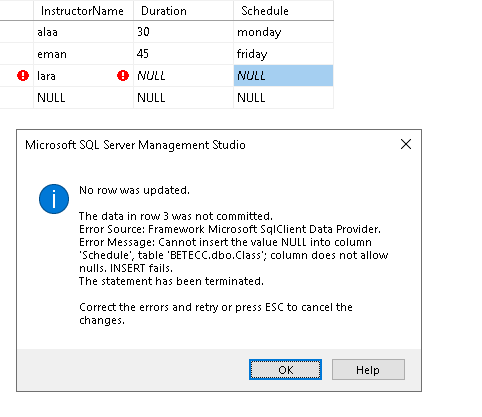
7- Network connectivity and infrastructure:

I checked the connection of the database with a strong and secure cloud database.

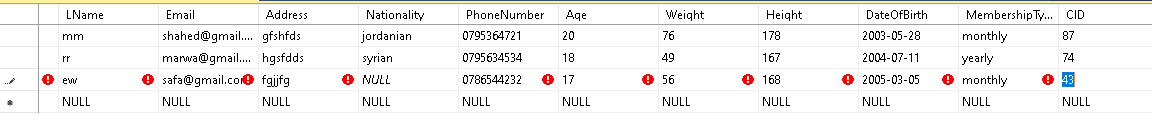
8- Backup and disaster recovery:

I made a backup for the database and tested it.

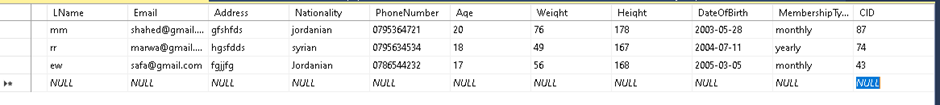
\*Test cases:

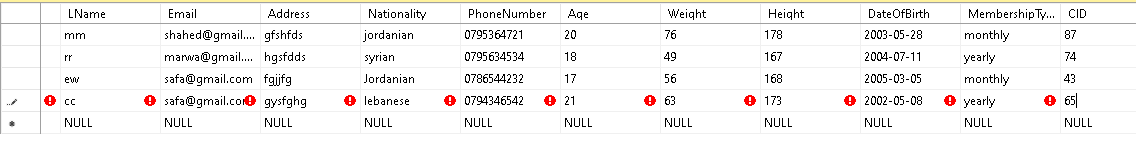
In this screenshot there is an error because the duration and schedule are null.

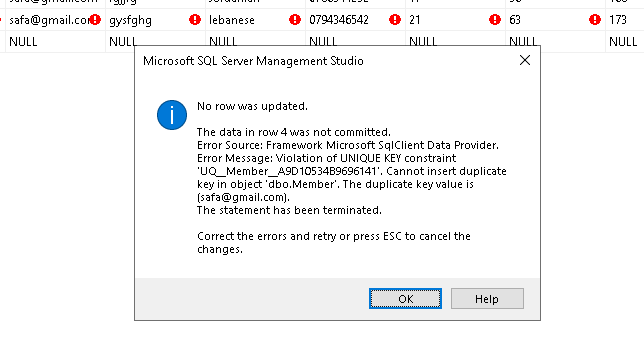
If just the duration column is null the same error will appear for duration.

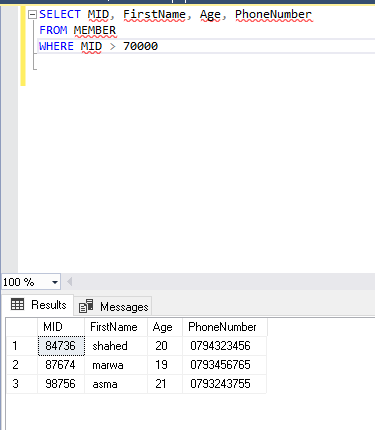
And if the schedule column is null it’s the same.  


In this screenshot the nationality is null, it should be “ Jordanian “ by default.





One of the constraints is not to register two users in the same email, so there is an error here.



This screenshot shows that the database running truly.

- How the selected test data can be used to improve the effectiveness of testing:

.Coverage: By selecting a diverse range of test data that covers various scenarios, edge cases, and boundary conditions, you can increase the coverage of your testing. This ensures that different aspects and functionalities of the system are thoroughly tested, identifying potential issues or vulnerabilities that might not be discovered with limited or uniform test data.

.Validation of Requirements: The selected test data should align with the user and system requirements. By using relevant and realistic test data, you can verify that the system behaves as expected and meets the specified requirements. This helps in ensuring that the system is developed and tested against the intended use cases.

.Error Detection: Test data that includes invalid or incorrect inputs can help in detecting errors, exceptions, or handling issues within the system. By intentionally introducing erroneous data, you can validate how the system handles such situations and whether appropriate error messages or fallback mechanisms are implemented.

.Performance and Scalability Testing: Test data that simulates different levels of workload, data volume, and concurrency can be used to assess the performance and scalability of the system. By measuring response times, throughput, and resource utilization with varying data sets, you can identify potential bottlenecks or performance degradation under different load conditions.

.Security Testing: Test data that includes various security scenarios, such as malicious inputs, data breaches, or unauthorized access attempts, can be used to evaluate the system's security mechanisms. By subjecting the system to different security tests, you can identify vulnerabilities, validate access controls, and ensure the system's robustness against potential threats.

.Regression Testing: Test data that represents typical and critical user scenarios can be used for regression testing. By retesting the system with the same or similar data after making changes or fixing issues, you can ensure that the modifications or fixes did not introduce new defects or regressions.

.Usability Testing: Realistic and representative test data allows you to assess the usability of the system. By analyzing how users interact with the system using the provided data, you can identify potential usability issues, such as confusing interfaces, cumbersome workflows, or data entry challenges.

.Data Integrity and Data Validation: Test data should cover different data types, ranges, and formats to validate data integrity and proper data validation mechanisms within the system.

.This helps in ensuring that data is correctly stored, processed, and retrieved without any data corruption or inconsistencies.

Q.8.

User Requirements:

-Users should be able to add and edit personal details to their profiles, including their name, email address, country, phone number, age, weight, and date of birth. ( in the visual studio interface using update button)

-During the registration process, members should be allowed to choose their membership type (monthly or yearly). (in the interface design in the first task and in the visual studio interface)

-Members ought to be able to monitor their accomplishments, measurements, and weight as well as their overall development. (in the query and visual studio interface using update button).

-Members should be able to plan and reserve lessons using the system according to their preferences and availability. (in the query and visual studio interface using update button).

-Members should be given recommendations for classes that complement their exercise routines and fitness objectives. (in the interface design in the first task)

-For a smooth member experience, the system should offer a user-friendly interface with simple navigation and intuitive design. ( in the interface design in the first task and in the visual studio interface)

-Members should be able to choose their privacy settings and have safe access to their personal information. (in the interface design in the first task)

System requirements:

-To guarantee seamless operation and no downtime, the cloud-based database system has to be scalable and capable of managing heavy loads during periods of peak demand .(I deployed the database of the system on AWS to be scalable and capable of managing heavy loads)

-To guarantee dependable data transfer and reduce downtime, the system should have strong network connectivity and a solid infrastructure. .(I deployed the database of the system on AWS to have strong network connectivity and a solid infrastructure)

-To safeguard member data, stop illegal access, and guarantee the privacy and confidentiality of sensitive information, a thorough security system should be put in place.( I applied a password for the database to be secure)

-Access to member profiles, equipment specifications, and class information should be quick and accurate thanks to the database system's excellent data storage and retrieval capabilities.

-In order to protect data integrity and enable speedy restoration in the event of any system failures or data loss occurrences, the system should include backup and disaster recovery procedures. (for the last 2 requirements I made a backup for the database, although AWS made a backup automatically).

Based on the assessment of the Nitrogen Gym cloud database system, here are some recommendations and suggestions to improve its effectiveness:

-Scalability and Performance Optimization:

.Evaluate the system's scalability to handle increasing loads and consider implementing auto-scaling mechanisms to dynamically allocate resources based on demand.

.Optimize database queries and indexes to improve response times and reduce latency.

.Implement caching mechanisms to reduce the load on the database and improve overall performance.

-Security Enhancements:

.Implement strong authentication and authorization mechanisms, such as multi-factor authentication and role-based access control, to ensure secure access to the system.

.Regularly update and patch the system's software and infrastructure components to address any known vulnerabilities.

.Monitor and log system activities to detect and respond to any potential security breaches in a timely manner.

-Disaster Recovery and Backup Strategy:

.Review and enhance the disaster recovery plan to ensure the system can recover quickly in case of any failures or data loss incidents.

.Regularly perform backups of the database and test the restoration process to ensure data integrity and availability.

-Data Validation and Integrity:

.Implement robust data validation mechanisms at the application layer to prevent invalid or malicious data from entering the database.

.Enforce referential integrity constraints and foreign key relationships to maintain data consistency.

-Monitoring and Alerting:

.Implement a comprehensive monitoring system to track the performance, availability, and health of the database system.

.Set up alerts and notifications to proactively detect and address any performance issues or anomalies.

-Documentation and Training:

.Provide clear and comprehensive documentation on the system's architecture, configuration, and usage guidelines.

.Conduct regular training sessions for system administrators and users to ensure they are familiar with best practices and security measures.

-Regular Audits and Penetration Testing:

.Perform periodic security audits and penetration testing to identify any vulnerabilities or weaknesses in the system.

.Address any identified issues promptly to maintain a secure environment for the database system.

-Continuous Improvement and Feedback:

.Establish a process for collecting and analyzing user feedback to identify areas for improvement.

.Continuously seek opportunities to enhance the system's functionality, performance, and user experience.

Q.9. Basing the implemented cloud database on access times, security of data, and cost has several ramifications that need to be considered and evaluated:

-Access Times:

Cloud databases generally offer good accessibility and fast response times. By utilizing a cloud-based solution, users can access the database from anywhere with an internet connection, facilitating convenience and flexibility.

The performance and access times of the cloud database can be optimized by implementing proper indexing, caching mechanisms, and load balancing techniques. This ensures efficient data retrieval and minimizes latency.

-Security of Data:

Cloud service providers typically have robust security measures in place to protect data. This includes physical security of data centers, encryption of data in transit and at rest, access controls, and regular security audits.

However, it's important to implement additional security measures specific to the application and database, such as strong authentication, encryption of sensitive data fields, and regular monitoring for any unauthorized access attempts.

Regularly reviewing and updating security protocols, staying informed about emerging threats, and ensuring compliance with relevant data protection regulations are crucial to maintaining data security.

-Viability Due to Cost:

Cloud databases offer cost-effective solutions compared to on-premises databases, as they eliminate the need for upfront infrastructure investments and provide flexibility in scaling resources based on demand.

However, it's important to carefully manage costs associated with cloud databases. This includes optimizing resource allocation, selecting appropriate service tiers, and monitoring usage to avoid unexpected expenses.

Regularly reviewing the database requirements and usage patterns can help identify opportunities for cost optimization, such as rightsizing resources or adopting reserved instances for predictable workloads.

-Data Backup and Disaster Recovery:

Cloud databases typically offer built-in backup and disaster recovery mechanisms. Regularly backing up data and testing the restoration process is essential to ensure data integrity and availability.

Consider the costs associated with data backup and disaster recovery, including storage fees and potential data transfer costs in case of recovery from a different region or provider.

Evaluate the Recovery Time Objective (RTO) and Recovery Point Objective (RPO) requirements of the database system to determine the appropriate backup and recovery strategy.

-Vendor Lock-in:

Adopting a cloud-based database solution may introduce a level of vendor lock-in, as migrating data and applications to a different provider or on-premises infrastructure can be challenging and time-consuming.

Evaluate the long-term plans and goals of the organization to ensure that relying on a specific cloud provider aligns with the business strategy.

Implement strategies for data portability and consider adopting multi-cloud or hybrid cloud architectures to mitigate the risk of vendor lock-in.

\*To ensure the ongoing suitability of the Nitrogen Gym cloud database system, the following improvements and suggestions can be implemented:

Continuous Performance Monitoring: Regularly monitor the performance of the cloud database system to identify any bottlenecks or issues that may affect access times. Utilize monitoring tools and performance metrics to track database performance, query execution times, and resource utilization. This will help identify areas for optimization and ensure a smooth user experience.

Scalability Planning: As the user base and data volume grow, it's important to plan for scalability. Evaluate the current system architecture and consider implementing horizontal scaling by distributing the database load across multiple instances or using sharding techniques. This will ensure that the system can handle increasing demands without compromising performance.

Regular Security Audits: Conduct regular security audits to identify potential vulnerabilities and ensure compliance with industry standards and regulations. Perform penetration testing and vulnerability assessments to identify any weaknesses in the system. Stay updated with security best practices and apply patches and updates promptly to mitigate security risks.

Disaster Recovery and Backup Strategy: Review the existing disaster recovery and backup strategy to ensure data integrity and availability. Implement automated backup processes with regular intervals and test the restoration process to ensure data can be recovered successfully. Consider adopting a multi-region backup strategy for enhanced data redundancy and protection against regional outages.

Data Encryption: Enhance data security by implementing encryption techniques at rest and in transit. Utilize encryption mechanisms provided by the cloud service provider or implement additional encryption layers within the application to protect sensitive data. This will ensure that even if unauthorized access occurs, the data remains secure.

Cost Optimization: Continuously monitor and optimize costs associated with the cloud database system. Review the usage patterns and resource allocation to identify opportunities for cost savings. Consider utilizing reserved instances or spot instances for predictable workloads and leverage auto-scaling to dynamically adjust resources based on demand. Implement effective resource tagging and tracking to identify cost centers and optimize resource allocation.

Regular System Updates and Patch Management: Stay up to date with the latest patches, updates, and security fixes provided by the cloud service provider. Establish a patch management process to ensure timely application of updates to address any security vulnerabilities or performance improvements.

User Feedback and Continuous Improvement: Gather feedback from users and stakeholders to identify areas for improvement. Implement a feedback mechanism or user surveys to collect insights and suggestions for enhancing the user experience and system functionality. Regularly review and prioritize user feedback to drive continuous improvement of the cloud database system.

**Plagiarism**

Plagiarism is a particular form of cheating. Plagiarism must be avoided at all costs and students who break the rules, however innocently, may be penalised. It is your responsibility to ensure that you understand correct referencing practices. As a university level student, you are expected to use appropriate references throughout and keep carefully detailed notes of all your sources of materials for material you have used in your work, including any material downloaded from the Internet. Please consult the relevant unit lecturer or your course tutor if you need any further advice.

**Student Declaration**

|  |
| --- |
| **Student declaration**  I certify that the assignment submission is entirely my own work and I fully understand the consequences of plagiarism. I understand that making a false declaration is a form of malpractice.  Student signature: Shahed Montaser Date:10/6/2023 |