Cybernetic Sabotage

Cybernetic Sabotage is an interesting thriller game that can be used to teach and test SQL concepts to students new to this topic.

In this game, the player plays the role of a leading cybersecurity analyst who has been called in after a major tech corporation, RoboTech Global, has experienced a series of strange occurrences within their AI-driven robotics division. There are fears of an inside job, corporate espionage, or even an AI gone rogue. Their mission is to use SQL to dive into the company's extensive databases and uncover the truth.

In classic fashion, this game comprises suspense and intrigue with an adventure role-play theme to introduce students to SQL concepts that grow in complexity gradually as the game progresses. It ends when the player has successfully used all the SQL concepts and written the relevant queries that eventually leads them to find the problem.

Concepts Covered

- Basic Query Structure of SQL Queries
- Additional Basic Operations
- Null Values
- Aggregate Functions
- Nested Subqueries
- Modification of the Database
- Join Expressions
- Views
- Integrity Constraints
- SQL Data Types and Schemas
- Given the schemas of a relation, create SQL queries using: SELECT, FROM, WHERE, EXISTS, NOT EXISTS, UNIQUE, NOT UNIQUE, ANY, ALL, DISTINCT, GROUP BY and HAVING.
- Set Operations
- Aggregate Functions
- Null Values Nested Subqueries

Database Schema

Employee: Employee ID, Name, Job Role, Division, Last Login Time

Robot: Robot ID, Model, Manufacturing Date, Status (functional, under repair, decommissioned), Last Software Update, Employee ID of the person who last updated the software

Log: Log ID, Employee ID, Action Description, Timestamp, and Robot ID

Incident: Incident ID, Description, Timestamp, Robot ID, Employee ID of the person who reported the incident

Access_Code: Access Code ID, Employee ID, Level of Access, Timestamp of the last use

Plot and Gameplay

1. The game begins with an urgent video call from the CEO of RoboTech Global. There's a situation - malfunctions with their AI-driven robots. You, a top cybersecurity analyst, have been hired to investigate. The player is then introduced to the database and the schema of each table.

2. The player is asked to list all incidents from the 'Incidents' table. The results reveal a series of strange malfunctions, all seem to involve the XJ-5 and QZ-4 robot models. [SELECT, FROM]

3. The player is asked to find the most recent incident involving these models. The data reveals a recent incident with an XJ-5 robot. [WHERE, ORDER, LIMIT]

4. The player needs to find out how many XJ-5 and QZ-4 model robots exist in the company. This requires a `JOIN` operation and the `COUNT` function. The results show a surprisingly high number of both models. [JOIN, COUNT, GROUP BY]

5. The player is asked to find out how many of these robots have been updated in the past one week, pointing towards possible tampering. A significant number of both models have been updated recently, indicating potential sabotage. [Set Operations, NULL]

6. The CEO wants to know which employees have updated these robots recently. The result reveals a list of employees. [Nested Subqueries]

7. The player is asked to mark all XJ-5 and QZ-4 robots as "under repair" in the database. [Modification of Database]

8. The player needs to identify the employee who reported the highest number of incidents. [Constraints]

9. The complexity of queries increases, and the player is encouraged to create a view to simplify their task. For instance, they may create a view that joins the 'Incidents' and 'Robots' tables to easily see all incidents associated with each robot model. [Views]

10. The player is tasked with identifying the source of the malfunctions. The player uses `GROUP BY` and `HAVING` to find models of robots that have more than a certain number of incidents, and `DISTINCT` to remove duplicate entries. The results show that the XJ-5 and QZ-4 models are indeed the common thread in these incidents. [EXISTS, HAVING, DISTINCT]

11. The player is asked to create a new table that records the repair status of all robots. They'll need to define the schema for this table, including appropriate data types for each column. [Data Types, Schema]

12. The player is then asked to insert/update repair records for all XJ-5 and QZ-4 robots into the new table. [Database Modification]

13. The player is tasked with finding the last employee who updated the software of the malfunctioning robots. [Subqueries, Sets]

14. The player finds out that there's a common thread - an employee who had last updated the malfunctioning robots. The results point to an employee named "Alex Mercer". [Join, Subqueries/Nested Queries]

15. The CEO is alarmed by this revelation and asks the player to gather all evidence against Alex Mercer. This would involve joining multiple tables and gathering data related to Alex's access logs, incidents reported, and robots he had access to. The player would use all SQL concepts learned from the game to gather this evidence. []

16. The game ends with a dramatic video call with Alex Mercer, where the player presents the evidence. Alex, cornered, confesses his actions, revealing his motives as a disgruntled employee. The player is congratulated for their diligent analysis and receives an offer for a co-op position at RoboTech Global's Vancouver office.

Progress So far

We are thrilled to provide you with an update on the remarkable strides we've made in the ongoing development of the enthralling "Cybernetic Sabotage" game. Our collective efforts have culminated in a captivating and visually stunning user interface that immerses players into a world akin to the Matrix hacker ambiance. Allow us to take you through the intricate details of this remarkable journey thus far.

The Inaugural Impression:

Upon accessing the game, players are greeted with an initial game screen adorned with an inviting "Start" button. This initial encounter sets the stage for the enthralling narrative that unfolds. Subsequent screens seamlessly weave an engaging storyline, leading players to the heart of the game—the Cybernetic Sabotage main game screen.

Immersive UI and Unveiling the Gameplay:

The Cybernetic Sabotage game screen exudes elegance and allure. The backdrop invokes a Matrix-inspired hacker aesthetic, instantly capturing players' attention. Central to the experience is the main panel—a dynamic canvas displaying the live implementation of the game. Here, players witness their queries materialize, an exhilarating representation of their actions.

The Art of Resilience: Error Handling

A masterstroke in our design lies in the panel's error handling mechanism. Should a player submit an erroneous query, the interface gracefully responds by displaying a pertinent error message. This mechanism nurtures a learning experience, guiding players toward mastery.

Navigational Arsenal: The Side Panel

A symphony of functionality adorns the side panel. Commanding the upper echelon are the Restart, Hint, and Settings buttons—a trio of empowerment. Directly beneath, resides a container, meticulously crafted to house vital game information: time, score, and queries solved. These elements form a compass to navigate the player's progress and aspirations.

The Trinity Enigma: Guiding Light of Gameplay

The bedrock of player guidance lies within the Trinity section—an epiphany for seekers. It is here that players glean questions and hints, seamlessly dovetailing with the overarching storyline. In the luminous realm of Trinity, players partake in an enchanting journey, unlocking the tapestry of their mission.

Hints as Currency: Unveiling the Veil

Within this realm, the judicious use of Hints becomes paramount. Each hint, a valuable currency, carries a specific cost—40, 60, or 80—tailored to their level. This strategic investment unveils pathways to enlightenment, an instrumental ally in deciphering the enigmatic landscape.

Score, Triumph, and Trial

The player embarks with an initial score of 150, accompanied by a progress bar at 10%. A triumph over each query propels the player toward victory, a calculated progression punctuated by newfound mastery of SQL concepts. Erroneous attempts, however, incur a cost, siphoning 10 points from the score. A tenuous balance ensues; should the score plummet to zero, victory slips through the grasp.

Embark on the Journey: Play "Cybernetic Sabotage"

Our game, "Cybernetic Sabotage," is seamlessly hosted on Vercel, ensuring efficient deployment and user accessibility. Explore the immersive experience at our GitHub repository: https://cybernetic-sabotage.vercel.app/

A Fusion of Artistry and Precision: Database Crafting

The foundation upon which Cybernetic Sabotage stands rests on the meticulous craftsmanship of its database. Employing the adept tools of DB Browser for SQLite, our development team has meticulously forged a database capable of housing, generating, and modifying game data seamlessly.

Elevating Code Quality: Linter Integration

To ensure unwavering code quality, we've instituted the assistance of a Linter. This diligent overseer maintains a vigilant watch, upholding a standard of excellence in our GitHub repository.

Profound Gratitude and Aspiration

Our gratitude extends to those who have supported us on this journey. To Professor Hazra, we've entrusted the true essence of the game—correct queries in the form of a txt file, a testament to our commitment to authenticity and academia.

A Glimpse Into the Forge: GitHub Repository

For those eager to delve deeper, our GitHub repository serves as a sanctum of development insights and progress. We cordially invite you to explore this repository, a living testament to our labor of love: https://github.com/lakshagarwal/CyberneticSabotage

Further Advancements/ Improvements

We are pleased to present a comprehensive overview of the recent developments in the Cybernetic Sabotage game, accompanied by insights into the strategic advancements and future improvements that have been identified to enhance both the learning experience for students and the potential for academic research.

Data Collection and Student Learning Insights:

With the successful realization of the Cybernetic Sabotage game, a significant avenue for data collection and analysis emerges. By integrating a login screen, we endeavor to acquire and store invaluable student data as they engage with the SQL learning journey offered by the game. This data repository will offer a treasure trove of insights into how students assimilate and apply various SQL concepts, thereby enabling informed pedagogical refinements.

Login Screen for Data Aggregation:

The introduction of a dedicated login screen represents a pivotal step toward our data collection objectives. By seamlessly integrating this screen into the game's architecture, we aim to capture essential student information that will contribute to a comprehensive understanding of their learning trajectories. This initiative aligns with our commitment to fostering evidence-based educational advancements.

Utilizing Data for Educational Insights:

The data harvested from student interactions within the game environment presents an exceptional opportunity to unravel the nuances of effective SQL learning. Our intention is to meticulously analyze the collected data, deciphering patterns, trends, and performance metrics. Such insights will empower educators with a deeper comprehension of pedagogically potent areas as well as potential areas of refinement.

Future Enhancement: Query Expansion and Difficulty Scaling:

In anticipation of an evolving educational landscape, our development team has conscientiously paved the way for future enhancements. By referencing the "main.js" file, developers can conveniently adjust the number and complexity of queries presented to players. This architectural flexibility equips us to tailor the learning experience to evolving curricular needs and varying student proficiency levels.

Iterative Development and Ongoing Refinement:

It is paramount to emphasize our dedication to iterative development and ongoing refinement. As we stride forward, we remain committed to cultivating the Cybernetic Sabotage game as a dynamic and perpetually improving educational instrument. The principles of agility, adaptability, and responsiveness underscore our approach to ensuring the game's enduring relevance.

In closing, we extend our gratitude for your unwavering support and guidance throughout this captivating odyssey. The strides we have taken underscore our collective determination to sculpt a transformative educational paradigm—one that synergizes immersive gameplay with profound pedagogical impact.

Should you wish to delve into the technical underpinnings or explore the intricacies of our developmental progress, we invite you to peruse our repository: https://github.com/lakshagarwal/CyberneticSabotage .

We remain at your disposal for any clarifications or further discussions.