|  |  |
| --- | --- |
| **Project Case** |  |
| ISYS6123 | ISYS6123003  Introduction to Database Systems |
| **Information Systems** | **E232-ISYS6123-MZ03-00** |
| ***Valid on*** *Even Semester Year 2022/2023* | **Revision 00** |

1. Kelompok tidak diperkenankan untuk:

*Members of the group are prohibited from:*

* + - Melihat sebagian atau seluruh jawaban kelompok lain,

*Seeing a part or the whole answer from other groups,*

* + - Menyadur sebagian atau seluruh jawaban dari buku, catatan, video, dan jenis referensi lainnya,

*Retell a part or the whole answer from books, notes, videos, and other references,*

* + - Menyadur sebagian atau seluruh jawaban dari internet,

*Retell a part or the whole answer from the internet,*

* + - Mengumpulkan jawaban yang tidak sesuai dengan tema soal,

*Submitting an answer with a different theme from the given case,*

* + - Melakukan tindakan yang menyebabkan jawaban dicontek oleh orang lain atau kelompok lain, baik disengaja maupun tidak disengaja,

*Doing action that could result the answer being copied by someone or other groups, intentionally or unintentionally,*

* + - Melakukan tindakan kecurangan lainnya.

*Committing other dishonest actions.*

1. Jika kelompok terbukti melakukan tindakan seperti yang dicantumkan pada butir ke-1, maka nilai mahasiswa dan/atau kelompok yang melakukan kecurangan, baik menyontek atau dicontek, akan dinolkan sesuai dengan peraturan yang berlaku.

*If it has been proven that a group has committed dishonest actions outlined in point 1 above, the whole groups related to the incident, regardless of which one copies or has their answer copied, will be issued a score of zero according to the regulation.*

1. Jawaban yang dapat diterima dan dinilai adalah jawaban yang dikumpulkan sebelum batas waktu yang telah ditentukan.

*The answer must be submitted before the designated deadline to be accepted and graded,*

1. Jawaban akan dinilai berdasarkan teknik atau metode yang diajarkan pada kelas praktikum dengan menggunakan software yang sudah ditentukan.

*The scoring will be based on the materials taught during the practicum classes using the designated software. Using different software than requested may result in your answer not being graded.*

1. Jika Anda tidak membaca peraturan ini, maka Anda dianggap sudah membaca dan menyetujuinya.

*By taking this exam, you agree to these regulations, regardless of whether you have read it or not.*

1. Persentase penilaian untuk matakuliah ini adalah sebagai berikut:

*The score will be distributed as follows:*

|  |  |  |
| --- | --- | --- |
| **Tugas Mandiri**  *Assignment* | **Proyek**  *Project* | **UAP**  *Final Exam* |
| 40% | 60% | - |

1. Perangkat lunak yang digunakan pada matakuliah ini adalah sebagai berikut:

*This course uses the following software:*

|  |
| --- |
| **Software**  *Software* |
| Microsoft Office 365  SQL Server Developer 2019  SQL Server Management Studio 18.9.1  Visual Paradigm Community Edition 16.3 |

1. Ekstensi file yang harus dikumpulkan untuk matakuliah ini adalah sebagai berikut:

*Your answers must be in the following file extensions:*

|  |  |  |
| --- | --- | --- |
| **Tugas Mandiri**  *Assignment* | **Proyek**  *Project* | **UAP**  *Final Exam* |
| SQL | SQL, VPP, Image Files (JPG / PNG) | - |

1. File yang harus dikumpulkan adalah keseluruhan jawaban beserta dengan aset yang digunakan (gambar, audio, video, dll) dan dokumentasi proyek yang berisikan link referensi aset dan penjelasan mengenai aplikasi yang dibuat (terlampir bersama dengan soal).

*Include other files that can support your project, such as: all files in your project, other files (image, audio, video, etc.) used in your project, \*.doc file (documentation of your project) that contains all pages in your project, reference links of additional files (image, audio, video, etc.) used in your project, the description about how to use your application, etc.*

## Soal

*Case*

**FuniMaZion**

**FuniMaZion** is a hot and trending web platform to watch many animes where the user could subscribe anime within a duration and staffs could purchase anime from studios.

Every **staff** that hired by **Funimazion** have a task to **purchase animes from studios**. Every staff must be following the **procedures** to become a staff, which are:

* Every staff hired must have a **personal information** like name, city, gender, email, password, and DOB (date of birth). Every staff has an **identification number** with the following format:

**SFXXX**

**X** à Number between 0 – 9

* Staff can **purchase** **one or many animes** from a studio
* Every **purchase** made with the studio have all the **information** about staff, studio, purchase date, and anime purchased. Every purchasehas an **identification number** with the following format:

**PUXXX**

**X** à Number between 0 – 9

* Every city where customer, staff, and studio lived has data of its name. Every city has an **identification number** with the following format:

**CIXXX**

**X** à Number between 0 – 9

* Every **anime** purchased from studio **have** its own **information** like genre, name, subscribe price per day, description, and minimum age. Every anime has an **identification number** with the following format:

**ANXXX**

**X** à Number between 0 – 9

* Every **anime genre** has **information** about the name. Every **genre** has an **identification number** with the following format:

**GEXXX**

**X** à Number between 0 – 9

* Every **sales** made with the customer have all the **information** about customer, sales date, sales method, anime sold, and the duration (day) of each anime. Every sales transaction has an **identification number** with the following format:

**SAXXX**

**X** à Number between 0 – 9

Every **customer** that wants to **subscribe anime** at **FuniMaZion** must be following the **transaction procedures**, those are:

* Every **customer** that wants to purchase a product must already completed personal **information** like name, city, gender, email, password, and DOB (date of birth). Every customer has an **identification number** with the following format:

**CUXXX**

**X** à Number between 0 – 9

* **Customer** can **purchase** **more than one anime** in every transaction.

Every **studio** that wants to **sell their anime** must be following the **sales procedures**, those are:

* Every **studio** that wants to sell their product must already completed personal **information** of name and city. Every studiohas an **identification number** with the following format:

**SDXXX**

**X** à Number between 0 – 9

* **Studio** can **sell** **more than one anime** in every sales transaction.

Here are some of the **constraints** requested by the funiMaZion.

* **Staff** and **customer gender** must be **either** “**Male**” **or** “**Female**” (without quote).
* **Customer email** must **end with** “**@gmail.com**” (without quote).
* **Customer password length** must more than or equal to 8.
* The **year of staff date of birth** must **before 2000**.
* **Anime minimum age** must **between 0 and 25.**
* **Sales method** must be either “**Credit**” **or “Debit”**.
* **All data must not** be **null**.

Now **FuniMaZion** still using manual management system to maintain the sales andpurchase transaction. You as a great **database analyst** are asked to design the database system to store data and maintain sales and purchase transactions. The tasks that you must do are:

* Create **Entity Relationship Diagram** to maintain transaction and purchase.
* Create a database system using **DDL syntax** that relevant with transaction and purchase.
* Create query using **DML syntax** to fill the tables in database systems with data based on the following conditions:
* **Master** **table** must be filled with **more than or equals 10 data**.
* **Transaction** **table** must be filled with **more than or equals 15 data**.
* **Transaction detail** **table** must be filled with **more than or equals 25 data**.
* For the **anime genre** table, the table must be filled with the following data:

|  |  |
| --- | --- |
| **Anime Genres** | |
| Drama | Action |
| Fantasy | Adventure |
| Romance | Comedy |
| Psychological | Horror |
| Sports | Mystery |

* Create query using DML syntax to **simulate the transactions** **process** for transaction and purchase. DML syntax to filldatabase and DML syntax to simulate the transactions process should be a **different query**.
* To support database management process in **funiMaZion**, mujak as the manager asked you to provide some **query** that resulting important data. The requirements that asked from him are:

1. Display StudioName and StaffCount (obtained from count how many distinct staff) live in the same city with the studio for every staff that was born after 1990 or Staff Password length less than 8.
2. Display CustomerName and CountAnimeNotFullfiledAgeCriteria (obtained from count anime the customer had subscribed) for every customer that is not fulfilled with the anime minimum age if today is January 1st, 2022, and SalesDuration is between 10 and 20.
3. Display StudioName, AnimeMinimumAge (obtained from the minimum of anime minimum age from all animes), AnimeMaximumAge (obtained from the maximum of anime minimum age from all animes), and LatestPurchaseDateQuarter (obtained from latest quarter of purchase date) for each studio name contains two or more words and anime description length is less than 10.
4. Display GenreName (obtained from uppercase of GenreName) and SaleCount (obtained from total sales count) and SaleDurationSum (obtained from the sum of total sale duration of animes ends with “ Days”) for each anime description that has at least 3 words and SaleCount is more than 5.
5. Display CustomerName (obtained from lowercase of CustomerName), AnimeName (obtained from uppercase of AnimeName), and AverageSpend (obtained from average of multiplication between animePricePerDay and salesDuration) for each customer that has lowercase or uppercase vocal alphabet in the first letter of its name and has AverageSpend more than the average of all AverageSpend of customers.

(**ALIAS SUBQUERY**)

1. Display CustomerName, SalesId (obtained by replacing first 2 characters of SalesId with 'Sales-'), and SalesDayOfYear (obtained from day of year from SalesDate) for each Sale transaction that has the cheapest anime and occurs in March, May, or July.

(**ALIAS SUBQUERY**)

1. Display CustomerName, CustomerDOB (obtained from formatting Customer to 'Dayname DD-Month-YY'), and TotalSpending (obtained from combining the sum of SalesDuration and AnimePricePerDay with ' Rupiah') for each customer born in the first half of the month (day of CustomerDOB is less than or equal 15) and has subscribed all kind of genre of anime.

(**ALIAS SUBQUERY**)

1. Display StaffName, StaffId (obtained from replacing 'SF' from Staffid to 'Staff '), StaffPassword (obtained from reversing staffPassword for security reason), and LatestPurchaseYear (obtained from the latest year of purchase transaction) for each staff that had handled the most purchase transactions with different studio and each studio name doesn’t contain any special character (special characters is characters beside lowercase or uppercase alphabet, numeric, and space) in its name.

(**ALIAS SUBQUERY**)

1. Create view CustomerData that displays CustomerName and CustomerEmailName (obtained from letters before '@' in CustomerEmail) for each customer that has average of SalesDuration more than 10 and never watch 2 or more animes with the same genre.
2. Create view AnimesCustomerMayWatch that displays CustomerName, CouldWatchAnimeCount (obtained from count anime), and AveragePrice (obtained by combining average of anime price per day ends with ' Rupiah') for every customer that fulfilled with the anime minimum age criteria if today is July 8th, 2021, and the average price is less than 18000.

**File that must be collected**:

1. Entity Relationship Diagram (.vsdx, .png)
2. Query to create the database system (.sql)
3. Query to insert data into tables (.sql)
4. Query to simulate the transactions processes (.sql)
5. Query to answer the 10 cases (.sql)