|  |  |
| --- | --- |
| **Project Case** |  |
| ISYS6169 | ISYS6279 | ISYS6280 | T0206 Database Systems |
| **Information Systems** | **O182-ISYS6169-PI03-00** |
| ***Valid on*** *Odd Semester Year 2019/2020* | **Revision 00** |

1. Seluruh kelompok tidak diperkenankan untuk:

*The whole group is not allowed to:*

* + - Melihat sebagian atau seluruh proyek kelompok lain,

*Seeing a part or the whole project from other groups*

* + - Menyadur sebagian maupun seluruh proyek dari buku,

*Adapted a part or the whole project from the book*

* + - Mendownload sebagian maupun seluruh proyek dari internet,

*Downloading a part or the whole project from the internet,*

* + - Mengerjakan soal yang tidak sesuai dengan tema yang ada di soal proyek,

*Working with another theme which is not in accordance with the existing theme in the matter of the project,*

* + - Melakukan tindakan kecurangan lainnya,

*Committing other dishonest actions,*

* + - Secara sengaja maupun tidak sengaja melakukan segala tindakan kelalaian yang menyebabkan hasil karyanya berhasil dicontek oleh orang lain / kelompok lain.

*Accidentally or intentionally conduct any failure action that cause the results of the project was copied by someone else / other groups.*

1. Jika kelompok terbukti melakukan tindakan seperti yang dijelaskan butir 1 di atas, maka **nilai kelompok** yang melakukan kecurangan (menyontek maupun dicontek) akan di – **NOL** – kan.

*If the group is proved to the actions described in point 1 above, the score of the group which committed dishonest acts (cheating or being cheated) will be “Zero”*

1. Perhatikan jadwal pengumpulan proyek, segala jenis pengumpulan proyek di luar jadwal tidak dilayani.

*Pay attention to the submission schedule for the project, all kinds of submission outside the project schedule will not be accepted*

1. Jangan lupa untuk melihat kriteria penilaian proyek yang ditempel di papan pengumuman, atau tanya asisten anda.

*Don’t forget to look at the project assessment criteria that posted on the announcement board, or ask your teaching assistant.*

1. Persentase penilaiaan untuk matakuliah ini adalah sebagai berikut:

*Marking percentage for this subject is described as follows:*

|  |  |  |
| --- | --- | --- |
| **Tugas Mandiri**  *Assignment* | **Proyek**  *Project* | **UAP**  *Final Exam* |
| 30% | 30% | 40% |

1. Software yang digunakan pada matakuliah ini adalah sebagai berikut:

*Software will be used in this subject are described as follows:*

|  |
| --- |
| **Software**  *Software* |
| Microsoft SQL Server Enterprise 2016  Microsoft Office 365 (Word, Excel)  Microsoft Office Visio 2013 |

## Ekstensi file yang harus disertakan dalam pengumpulan tugas mandiri dan proyek untuk matakuliah ini adalah sebagai berikut:

*File extensions should be included in assignment and project collection for this subject are described as follows:*

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

## Soal

*Case*

**GameRika**

**GameRika** is the best website for selling games managed by your friend, Rika. Rika and her friends manage all of activities that belongs to **GameRika**. Userswho are registered to the site can purchase games online through the site.

Every user that wants to buy a game in **GameRika** must follow the **sales transaction procedures**, which are:

* User that wants to buy a game must have a personal information like nickname, full name, email, country, and description. Every **user** has an identification number with the following format:

“USRXXX”

X => number between 0 – 9

* Every **sales transaction** made have all the information about the user, the transaction date, the game(s) purchased, and the quantity of each game. Every **sales transaction** has an identification number with the following format:

“SALXXX”

X => number between 0 – 9

* User can purchase **more than one game** in a sales transaction

Every game to be sold in **RikaGame** must be registered the with the following information, which are:

* Game to be sold must have a complete information like the developer, the publisher, title, description, release date, price, and the genre(s). Every **game** has an identification number with the following format:

“GAMXXX”

X => number between 0 – 9

* Every game **developer** has a complete information like name, email, country, address, phone, and an identification number with the following format:

“DEVXXX”

X => number between 0 – 9

* Every game **publisher** has a complete information like name, email, country, address, phone, and an identification number with the following format:

“PUBXXX”

X => number between 0 – 9

* Every game may have **more** **than one** genre.
* Every **genre** has a complete information like name and an identification number with the following format:

“GENXXX”

X => number between 0 – 9

Every user that is already registered in **RikaGame** can also give a **review** to a game by following these procedures:

* User can only leave **one** review on every game.
* Every game **review** contains all information about the user, the game, recommendation status (recommended or not recommended), review content, and review date.

**Notes:**

* User nickname must be more than 5 characters.
* User email must have ‘@’ character (without quote).
* Publisher email must have ‘@’ character (without quote).
* Publisher phone must be numeric.
* Developer email must have ‘@’ character (without quote).
* Developer phone must be numeric.
* Game description must be more than 20 characters.
* Game release year must be greater than 1997.
* Review content must be more than 20 characters.

Now **GameRika** is still using manual management system to maintain the **sales transactions**. You as her precious friend wants to help **GameRika** to create a database system that can store data and maintain the **sales transactions** and **review transactions**. The tasks that you must do are:

1. Create Entity Relationship Diagram to maintain **sales transactions** and **review transactions**.
2. Create a database system using DDL syntax that relevant with **sales** **transactions**. The database system must include database and tables with the required procedures.
3. 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 to 10 data.
* **Transaction** table must be filled with more than or equals to 15 data.
* **Transaction detail / Mapping table** must be filled with more than or equals 25 data.

1. Create query using DML syntax to simulate the transactions process for **sales** **transactions** and **review transactions**.

**Note**: DML syntax to **fill database** and DML syntax to **simulate** the **transactions process** should be a **different query**.

1. To support database management process in **GameRika**, Rika asked you to provide some query that resulting important data. The requirements that asked from her are:
2. Display GameId, GameTitle, and Total Review (obtained from the total review of that game and ended with ‘ review(s)’) for each game that is reviewed by as user who lived in ’Singapore’ or ’Indonesia’ and published by a publisher which ID is ‘PUB002’.
3. Display GenreId, GenreName, and Total Game (obtained from the count of game) for every game which is developed by a developer which ID is between ‘DEV003’ and ‘DEV009’ and the game is released in even month.
4. Display DeveloperId, DeveloperName, Local Phone (obtained by replacing DeveloperPhone first character into ‘+62’), Games Sold (obtained from the sum of quantity), and Total Transaction (obtained from the count of transaction) for every developer whose ID between ‘DEV004’ and ‘DEV008’ and the Games Sold is less than or equals to 5.
5. Display UserNickname, User Country (obtained from UserCountry in uppercase format), Total Game Purchased (obtained from the sum of quantity), and Game Owned (obtained from the count of game) for every user which ID is ‘USR004’ or ‘USR005’ and purchased a game that is released on odd day of the month.
6. Display Numeric User Id (obtained from the last 3 characters of UserId), Nickname (obtained from UserNickname in uppercase format), and UserCountry for every user who purchased a game with a quantity higher than average of all game quantity sold and the user nickname contains ‘k’.

(**alias subquery**)

1. Display GameId, GameTitle, PublisherName, and Publisher Email (obtained from the characters before ‘@’ in PublisherEmail) for every game which is sold with quantity more than the maximum quantity of all transaction that occurred on the 22nd day of that month.

(**alias subquery**)

1. Display DeveloperId, DeveloperName, Game Title (obtained from GameTitle in lowercase format), and Total Genre (obtained from the count of genre and ended with ‘ genre(s)’) for every developer who created a game on a day greater than the maximum day of every game that is released on April.

(**alias subquery**)

1. Display UserNickname, User First Name (obtained from the first word of UserFullName), Sales Date (obtained from SalesDate in ‘dd mon yyyy’ format), and Total Quantity (obtained from the sum of quantity) for every user who purchased a game with a quantity greater than the maximum quantity of all transaction that occurred on 20th day of the month.

(**alias subquery**)

1. Create a view named ‘**CustomUserViewer**’ to display UserId, UserNickname, Minimum Quantity (obtained from minimum of quantity sold), and Maximum Quantity (obtained from maximum of quantity sold) for every transaction that occurred on the 25th day of the month and the user’s nickname contains ‘r’.
2. Create a view named ‘**CustomPublisherViewer**’ to display PublisherName, GameTitle, Release Date (obtained from GameReleaseDate in ‘dd mon yyyy’ format), Total Purchase (obtained from the sum of quantity), and Minimum Purchase (obtained from the minimum of quantity) for every publisher which released a game from a developer in the same country and the game was released in April.

**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)

**Here are the rules that you must follow to create your project:**

1. Use appropriate software for this subject based on **Sistem Praktikum** that can be downloaded from Binusmaya.
2. Use the techniques taught during practicum.
3. Collect appropriate files for this subject based on **Sistem Praktikum** that can be downloaded from Binusmaya.
4. Include the 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 the reference links of additional files (image, audio, video, etc.) used in your project