|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Practicum Final Exam – Odd Semester Year 2021/2022** | | | | | | |
| **Subject** | | | **ISYS6169001 – Database Systems** | | | Diagram, schematic  Description automatically generated |
| **Class** | **:** | **B401 / B501** | | **Start Date** | **: 12 January 2022** |
| **Lecturer** | **:** | **D4767 - Harkat Christian Zamasi, S.Kom., M.M.S.I** | | **Start Time** | **: 15:20** |
| **End Date** | **: 12 January 2022** |
| **End Time** | **: 17.20** |

**PERATURAN UJIAN:**

*Exam Regulations:*

* Mahasiswa tidak diperbolehkan berdiskusi dan/atau bekerja sama dengan peserta ujian lainnya

*Student is not allowed to discuss and/or work together with other exam participants*

* Mahasiswa tidak diperbolehkan untuk membuka dan menyalin dari **BUKU** atau **CATATAN**, **VIDEO** dari pengajar (recording kelas, VBL, Youtube, dsb) dan **REFERENSI** lainnya

*Student isn't allowed to open and copy from any resources such as notes, videos (class recording, VBL, Youtube, etc) and other references*

* Mahasiswa tidak diperbolehkan membuka dan menyalin jawaban dari internet (google, stackoverflow, dsb)

*Student isn't allowed to open and copy answer from the internet (google, stackoverflow, etc)*

* Asisten **BERHAK** memberi nilai 0 **(NOL)** bagi peserta ujian yang melakukan segala bentuk kecurangan

*Assistant is able to give 0 (ZERO) score for exam participant who does any cheating actions*

* Kumpulkan jawaban tepat pada waktunya, apabila terlambat mengumpulkan maka jawaban tidak akan dikoreksi dan nilai mahasiswa adalah 0

*Submit the answer on time, if not, then the answer will not be checked, and the students will receive 0 (ZERO)*

* Bila Anda tidak membaca peraturan ini, maka Anda dianggap telah membaca dan menyetujuinya

*If you have missed to read these regulations, so you are considered to have read and agreed on it*

Logo, company name

Description automatically generated

**SOFTWARE YANG DIGUNAKAN:**

*Software will be used:*

* SQL Server Developer 2019
* SQL Server Management Studio 18.9.1

**FILE YANG DIKUMPULKAN:**

*File must be collected:*

* SQL

**PERHATIAN!**

*Attention!*

* Bagi yang mengerjakan tidak sesuai dengan soal, maka akan diberikan nilai **NOL (0)**

*For those who do not work in accordance with the exam case will be marked as* ***ZERO (0)***

* Bagi yang mengerjakan tidak sesuai dengan software dan versi yang telah ditetapkan, maka akan tetap dikoreksi dengan software dan versi yang telah ditetapkan

*For those who do not work in accordance with the software and specific version will be corrected by the predefined software and version*

* Kompres semua jawaban yang akan diunggah. Pastikan format pengumpulan nama file dan ekstensi sesuai dengan format berikut: **[NIM]-[NAMA].zip**

*Compress all file that will be uploaded. Make sure the format for collecting file name and extension according to the following format:* ***[NIM]-[NAME].zip***

**Tabel Relasional**

*Relational Table*

**Planet Tire**

![Diagram

Description automatically generated]()

**Soal**

*Case*

1. Display **CustomerId**, **CustomerName**, **CustomerPhone**, and **CustomerEmail** for every Customer whose **name**’s **contains** **‘Filip’**.

**(like)**



1. Display **StaffId**, **StaffName**, **StaffPhone**, **StaffEmail** for every transaction that take place on **Monday**.

**(join, datename, weekday)**

**Text

Description automatically generated**

1. Display **TypeId** (obtained from the **last 3 characters** from TypeId), **TypeName**, **TypeDurability**, and **Total Stock** (obtained from **total** of **tire’s Stock**) for every tire which **type** is **'Off Road'**.

**(right, sum, join, like)**

****

1. Display **TypeId**, **TypeName**, **Highest Tire Sold** (obtained from the **highest quantity** of **tires sold**), and **Total Stock** (obtained from the **total** of **tire’s Stock**) for every tire which **type** is **'Off Road'**. Then, **combine** it with **TypeId**, **TypeName**, **Highest Tire Sold** (obtained from the **highest quantity** of **tires sold**), and **Total Stock** (obtained from **total** of **tire’s Stock**) for every tire’s **Total Stock** that is **more than 2000**. After that, **order** the results by **Total Stock** **descendingly**.

**(max, sum, like, order by, union)**

**Graphical user interface, table

Description automatically generated**

1. Display **TypeId**, **TypeName** in **uppercase letter**, **TireName**, **DateCreated** in **'Mon dd, yyyy'** format, **Stock**, and **Quantity** for every tire whose **name** is **'ICESTONE', 'TIRELLI',** or **'DEVELOP TIRES'** and the **number of tires sold** **more than 4**.

**(upper, convert, in)**

**A screenshot of a computer

Description automatically generated with medium confidence**

1. Display **SalesId**, **SalesDate** in **'dd mon yyyy'** **format**, **TireName**, **TypeName**, and **Quantity** (with **‘ Pc(s)’** added in the **end** **of the quantity**) for every transaction that take place in **2020** and the tire’s **TypeDurability** is **more than and equals the average of all tire’s TypeDurability**.

**(cast, convert, alias subquery, avg, year)**

**Graphical user interface, text, application, table

Description automatically generated with medium confidence**

1. Create a **view** named ‘**StaffSalesRecap’** to display **StaffId**, **SalesId**, **StaffName**, **StaffEmail**, and **SalesDate** in **'dd/mm/yyyy'** format for every staff who handle the transaction **before 2019**.

**(create view, convert, year)**

**Graphical user interface

Description automatically generated**

1. Create a **view** named ‘**TireSoldSummary’** to display **TypeId** (obtained from **replacing 'TY' to 'Type** '), **TypeName**, **TypeDurability**, **Highest Sold Quantity** (obtained from the **highest quantity of the tire’s sold**), and **SalesDate** for every tire’s **TypeDurability** is **more than 70** and the transaction is taken place **on August**.

**(create view, replace, max, month)**

**Chart

Description automatically generated**

1. Add a column named ‘**TypeMaterial’** on **MsTireType** table with **varchar(50)** data type. Then, **add a constraint** on **MsTireType** table to check whether the **TypeMaterial** is **'Natural Rubber'** or **'Synthetic Rubber'.**

**(alter table, add, add constraint, check)**

**Before:**

**Table

Description automatically generated**

**After:**

**Table

Description automatically generated**

1. **Update** Staff data by **increasing StaffSalary** **by 1.000.000** for every tire’s **sold** **greater than or equals 8** and for each **StaffSalary** **under 5.000.000**.

**(update)**

**Before:**

**Text, table

Description automatically generated**

**After:**

**Graphical user interface, text, table

Description automatically generated with medium confidence**