

Beheer database

1. Maak voor de database “Poortwachter” een gebruiker aan onder de naam “Kandidaat” en zorg er voor dat hiermee vanaf elke host uitsluitend “Selects” en “Inserts” uitgevoerd mogen worden. Maak een screenshot van de aanpassingen en neem deze op in je verslag. Voor het opstellen van je verslag maak je gebruik van “Bijlage 3.3 Sjabloon verslag SQL”.

Query's maken

1. Query 1 : Laat met een query alle velden van tabel “GescandeData” zien.
`SELECT * FROM GescandeData`
2. Query 2 : Laat met een query de velden “Kenteken” en “Merk” uit tabel “GescandeData” zien.
`SELECT Kenteken, Merk FROM GescandeData`
3. Query 3 : Laat met een query de velden “Kenteken”, “Merk” en “Cilinderinhoud” zien uit tabel “RDW”.
`SELECT Kenteken, Merk, Cilinderinhoud FROM RDW`
4. Query 4 : Maak een query van de velden “Kenteken”, “Merk” uit de tabel “GescandeData” waarvan het kenteken niet in de tabel “RDW” te vinden is.
`SELECT Kenteken, Merk FROM GescandeData
WHERE Kenteken NOT IN (SELECT Kenteken FROM RDW)`
5. Query 5 : Doe een SQL-insert van het kenteken “1SZN01”, het merk “Suzuki” en de handelsbenaming “Alto Scantest” in tabel “GescandeData”.
`ALTER TABLE GescandeData
MODIFY SupplierID INT DEFAULT 0

INSERT INTO GescandeData (Kenteken, Merk, Handelsbenaming)
VALUES ('1SZN01', 'Suzuki', 'Alto Scantest')`
6. Query 6 : Laat via een query de velden “Kenteken” en “Merk” uit tabel “RDW” zien waarbij de “Cilinderinhoud” meer is dan de gemiddelde “Cilinderinhoud” van alle auto’s (waarbij de gemiddelde cilinderinhoud natuurlijk in de query berekend moet worden).
`SELECT Kenteken, Merk FROM RDW
WHERE Cilinderinhoud > (SELECT AVG(Cilinderinhoud) FROM RDW)`

Verslag maken

Je maakt een verslag volgens "Bijlage 3.3 Sjabloon verslag SQL". In je verslag neem je de screenshots van de door jou gemaakte aanpassing van de rechten van de gebruiker op en de screenshots van de query's en de queryresultaten (voor zover dat past op een uitsnede).

1. De SQL query "SELECT * FROM GescandeData" selecteert alle velden van de tabel "GescandeData" in de database. De sterretje (*) geeft aan dat alle kolommen van de tabel moeten worden geselecteerd. Het resultaat van de query is een lijst van alle rijen uit de "GescandeData" tabel, met elke kolom die wordt weergegeven als een aparte kolom in het resultaat.
2. De SQL query "SELECT Kenteken, Merk FROM GescandeData" is een selectie-query die de velden "Kenteken" en "Merk" uit de tabel "GescandeData" opvraagt. De query zal de geselecteerde gegevens in een tabelvorm weergeven, met de kolommen "Kenteken" en "Merk" en de bijbehorende waarden voor elk record in de tabel "GescandeData". Dit kan nuttig zijn bijvoorbeeld voor het analyseren van gegevens over gescande voertuigen.
3. Deze SQL query selecteert de velden "Kenteken", "Merk" en "Cilinderinhoud" uit de tabel "RDW". Dit zorgt ervoor dat alleen de informatie over deze specifieke velden wordt getoond in het resultaat. De query geeft dus alleen informatie over het kenteken, het merk en de cilinderinhoud van de voertuigen in de RDW database.
4. Deze SQL query selecteert de velden "Kenteken" en "Merk" uit de tabel "GescandeData" waarvan het kenteken niet in de tabel "RDW" te vinden is. Dit betekent dat alleen de records uit de tabel "GescandeData" worden geselecteerd waarbij het kenteken niet voorkomt in de tabel "RDW". De query gebruikt de NOT IN operator om te controleren of het kenteken van een record uit de tabel "GescandeData" niet voorkomt in de tabel "RDW". Als dit het geval is, wordt het record geselecteerd en getoond in de query resultaten. Dit kan nuttig zijn voor bijvoorbeeld het controleren van niet-geregistreerde voertuigen op een parkeerterrein.
5. De SQL-query begint met een ALTER TABLE statement om de kolom "SupplierID" in de tabel "GescandeData" te wijzigen en een standaardwaarde van 0 toe te wijzen. Dit betekent dat als er geen waarde wordt gegeven voor deze kolom tijdens een nieuwe insert, er automatisch een waarde van 0 wordt toegekend.

Vervolgens wordt er een INSERT INTO statement gebruikt om nieuwe gegevens toe te voegen aan de tabel "GescandeData". In deze query worden de waarden "1SZN01" voor het kenteken, "Suzuki" voor het merk en "Alto Scantest" voor de handelsbenaming geïnsert in de juiste kolommen van de tabel.

In samenvatting voegt deze SQL-query een nieuwe rij toe aan de tabel "GescandeData" met de gegevens van het kenteken "1SZN01", het merk "Suzuki" en de handelsbenaming "Alto Scantest". Tevens wordt de standaardwaarde van de kolom "SupplierID" op 0 gezet.

6. Deze SQL query selecteert de velden "Kenteken" en "Merk" uit de tabel "RDW". Er wordt vervolgens een filter toegepast waarbij alleen auto's met een "Cilinderinhoud" groter dan het gemiddelde van alle auto's in de tabel "RDW" worden getoond. Dit gemiddelde wordt binnen de query berekend door gebruik te maken van de AVG functie op het veld "Cilinderinhoud". Hierdoor wordt een lijst van auto's met een cilinderinhoud boven het gemiddelde getoond.