**Concept 1**

**Problem Statements**

|  |
| --- |
| **Problem Statement 1.1:**  Jimmy, from the healthcare department, has requested a report that shows  how the number of treatments each age category of patients has gone through in the year 2022. The age category is as follows, Children (00-14 years), Youth (15-24 years), Adults (25-64 years), and Seniors (65 years AND over). Assist Jimmy in generating the report. |

**Solution 1.1:**

ALTER TABLE patient ADD column Age INT;

UPDATE patient set age = DATE\_FORMAT(FROM\_DAYS(DATEDIFF('2022-01-01', dob)), '%Y');

SELECT age\_category, COUNT (patientID) AS 'Patient COUNT '

FROM (SELECT \*, CASE

WHEN age BETWEEN 0 AND 14 THEN 'Children'

WHEN age BETWEEN 15 AND 24 THEN 'Youth'

WHEN age BETWEEN 25 AND 64 THEN 'Adults'

ELSE 'Seniors'

END AS 'Age\_Category'

FROM patient ) AS q1

JOIN treatment USING(patientid)

WHERE year(date) = 2022

GROUP BY age\_category;

**Output 1.1:**

|  |  |
| --- | --- |
| Age\_Category | Patient Count |
| Adults | 1454 |
| Children | 812 |
| Seniors | 649 |
| Youth | 52 |

|  |
| --- |
| **Problem Statement 1.2:**  Jimmy, from the healthcare department, wants to know which disease is infecting people of which gender more often. Assist Jimmy with this purpose by generating a report that shows for each disease the male-to-female ratio.  Sort the data in a way that is helpful for Jimmy. |

**Solution 1.2:**

WITH patient\_COUNT AS

(SELECT diseaseid, gender, COUNT (patientid) AS Patients

FROM Person AS p

JOIN treatment AS t on t.patientid = p.personid

GROUP BY diseaseid, gender

ORDER BY diseaseid

)

SELECT diseaseid,diseasename, f.Patients, m.Patients

FROM disease

JOIN (SELECT diseaseid, Patients FROM patient\_COUNT WHERE gender = 'Female') AS f

USING(diseaseid)

JOIN (SELECT diseaseid, Patients FROM patient\_COUNT WHERE gender = 'Male') AS m

USING(diseaseid);

**Output 1.2:**

|  |  |  |  |
| --- | --- | --- | --- |
| diseaseid | diseasename | Patients | Patients |
| 1 | Alzheimer's disease | 95 | 173 |
| 2 | Amyotrophic lateral sclerosis | 106 | 165 |
| 3 | Anorexia nervosa | 96 | 177 |
| 4 | Anxiety disorder | 126 | 153 |
| 5 | Asthma | 101 | 144 |
| 6 | Atherosclerosis | 112 | 174 |
| 7 | Attention deficit hyperactivity disorder | 125 | 158 |
| 8 | Autism | 94 | 156 |
| 9 | Autoimmune diseases | 102 | 165 |
| 10 | Bipolar disorder | 114 | 166 |

|  |
| --- |
| **Problem Statement 1.3:**  Jacob, from insurance management, has noticed that insurance claims are not made for all the treatments. He also wants to figure out if the gender of the patient has any impact on the insurance claim. Assist Jacob in this situation by generating a report that finds for each gender the number of treatments, number of claims, and treatment-to-claim ratio. and notice if there is a significant difference between the treatment-to-claim ratio of male and female patients. |

**Solution 1.3:**

SELECT Gender, COUNT (treatmentID) AS 'Number of Treatments',

COUNT (claimID) AS 'No of Claims',

COUNT (treatmentID)/COUNT (claimID) AS 'Treatment to Claim Ratio'

FROM Person AS p

JOIN treatment AS t on t.patientid = p.personid

GROUP BY gender;

**Output 1.3:**

|  |  |  |  |
| --- | --- | --- | --- |
| Gender | Number of Treatments | No of Claims | Treatment to Claim Ratio |
| male | 6679 | 4287 | 1.558 |
| female | 4206 | 2676 | 1.5717 |

|  |
| --- |
| **Problem Statement 1.4:**  The Healthcare department wants a report about the inventory of pharmacies. Generate a report on their behalf that shows how many units of medicine each pharmacy has in their inventory, the total maximum retail price of those medicines, AND the total price of all the medicines after discount. Note: discount field in keep signifies the percentage of discount on the maximum price. |

**Solution 1.4:**

SELECT pharmacyid, PharmacyName, round(sum(quantity),0) AS 'Total Medicine',

round(sum(quantity\*maxprice),0) AS 'Total MRP of All Medicine',

round(sum(quantity\*maxprice\*(1-discount /100)),0) AS 'Total Price After Discount '

FROM keep

JOIN medicine USING(medicineID)

JOIN pharmacy USING(pharmacyID)

GROUP BY pharmacyid;

**Output 1.4:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| pharmacyid | PharmacyName | Total Medicine | Total MRP of All Medicine | Total Price After Discount |
| 3287 | Ally Scripts | 1037993 | 242673230 | 207645810 |
| 8184 | Apotheco | 2278817 | 978659351 | 796193118 |
| 8911 | Assured Rx | 1306242 | 950683603 | 856578606 |
| 4996 | Banks Apothecary | 994595 | 630995153 | 532609870 |
| 3703 | Bioplus Specialty | 1898455 | 870520226 | 722464875 |
| 2209 | Blink Health | 2305042 | 611615544 | 519736912 |
| 5960 | CareFirst | 631870 | 376846655 | 310478476 |
| 1624 | Caremark | 1996254 | 1255294071 | 1115830623 |
| 5866 | Carepoint | 2199862 | 586342803 | 494604364 |

|  |
| --- |
| **Problem Statement 1.5:**  The healthcare department suspects that some pharmacies prescribe more medicines than others in a single prescription, for them, generate a report that finds for each pharmacy the maximum, minimum and average number of medicines prescribed in their prescriptions. |

**Solution 1.5:**

SELECT pharmacyID, pharmacyName, min(medicine\_COUNT ) AS 'Min\_Prescribe',

max(medicine\_COUNT ) AS 'Max\_Prescribe',

round(avg(medicine\_COUNT ),0) AS 'Avg\_Prescribe'

FROM ( SELECT pharmacyID, prescriptionID, COUNT (medicineID) AS 'Medicine\_COUNT '

FROM prescription AS pr

JOIN contain AS c USING (prescriptionID)

GROUP BY pharmacyID, prescriptionID ) AS q1

JOIN pharmacy USING(pharmacyID)

GROUP BY pharmacyID

ORDER BY Avg\_Prescribe desc, Max\_Prescribe desc, min\_prescribe desc;

**Output 1.5:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| pharmacyID | pharmacyName | Min\_Prescribe | Max\_Prescribe | Avg\_Prescribe |
| 9403 | Pharmaca Integrative | 2 | 12 | 5 |
| 1891 | Welltrack | 1 | 12 | 5 |
| 2426 | Lifeshave Pharmacy | 3 | 11 | 5 |
| 2449 | Simple Meds | 1 | 11 | 5 |
| 2836 | Discount Drug Mart | 1 | 11 | 5 |
| 5947 | Union Center Pharmacy | 1 | 11 | 5 |
| 5976 | Pride Pharmacy | 1 | 11 | 5 |
| 8549 | Cash Saver Pharmacy | 1 | 11 | 5 |
| 9239 | ScriptSave | 1 | 11 | 5 |

**Solution 2**

**Problem Statements**

|  |
| --- |
| **Problem Statement 2.1:**  A company needs to set up 3 new pharmacies, they have come up with an idea that the pharmacy can be set up in cities where the pharmacy-to-prescription ratio is the lowest and the number of prescriptions should exceed 100. Assist the company to identify those cities where the pharmacy can be set up. |

**Solution 2.1 :**

SELECT city, COUNT (distinct pharmacyID)/COUNT (distinct prescriptionID) AS 'pharmacy\_to\_prescription'

FROM address

JOIN pharmacy USING(addressid)

JOIN prescription USING(pharmacyid)

GROUP BY city

HAVING COUNT (prescriptionID)>100

ORDER BY pharmacy\_to\_prescription

LIMIT 3;

**Output 2.1:**

|  |  |
| --- | --- |
| city | pharmacy\_to\_prescription |
| Worcester | 0.0137 |
| Panama City Beach | 0.014 |
| Glen Burnie | 0.0143 |

|  |
| --- |
| **Problem Statement 2.2:**  The State of Alabama (AL) is trying to manage its healthcare resources more efficiently. For each city in their state, they need to identify the disease for which the maximum number of patients have gone for treatment. Assist the state for this purpose. Note: The state of Alabama is represented as AL in Address Table. |

**Solution 2.2 :**

WITH Patients AS

( SELECT city, diseaseid, COUNT(patientID) AS Patient\_count FROM address

JOIN person USING(addressid)

JOIN treatment on person.personid = treatment.patientID

WHERE state = 'AL'

GROUP BY city, diseaseid

ORDER BY city )

SELECT city, diseaseid, diseaseName, Patient\_count FROM Patients P1

JOIN disease AS d USING(diseaseid)

WHERE patient\_count = (SELECT max(Patient\_count ) FROM Patients AS p2 WHERE

P1.city = P2.city)

ORDER BY patient\_count desc, city, diseaseName;

**Output 2.2:**

|  |  |  |  |
| --- | --- | --- | --- |
| city | diseaseid | diseaseName | Patient\_Count |
| Montgomery | 11 | Cancer | 28 |
| Montgomery | 22 | GuillainâBarrÃ syndrome | 28 |
| Montevallo | 36 | Schizophrenia | 2 |
| Indian Springs Village | 1 | Alzheimer's disease | 1 |
| Indian Springs Village | 10 | Bipolar disorder | 1 |
| Indian Springs Village | 19 | Diabetes mellitus type 2 | 1 |
| Indian Springs Village | 27 | Multiple sclerosis | 1 |
| Indian Springs Village | 32 | Parkinson's disease | 1 |
| Indian Springs Village | 36 | Schizophrenia | 1 |

|  |
| --- |
| **Problem Statement 2.3:**  The healthcare department needs a report about insurance plans. The report is required to include the insurance plan, which was claimed the most and least for each disease. Assist to create such a report |

**Solution 2.3 :**

WITH Insurence\_Count AS

( SELECT diseaseID, planName, COUNT(patientID) AS 'Plan\_Claimed'

FROM treatment

JOIN claim using(claimId)

JOIN insuranceplan using(UIN)

GROUP BY diseaseid, planName

ORDER BY diseaseid, planName )

SELECT diseaseID, planName, (plan\_claimed), row\_number() over(partition by diseaseID, Plan\_claimed) as 'Plan\_Count'

FROM ( (SELECT DiseaseID, planName, Plan\_claimed

FROM Insurence\_Count as IC1

WHERE Plan\_claimed = (SELECT MIN(Plan\_Claimed) FROM Insurence\_count IC2

WHERE IC1.DiseaseID= IC2.DiseaseID ))

UNION

( SELECT DiseaseID, planName, Plan\_claimed FROM Insurence\_Count AS IC1

WHERE Plan\_claimed = (SELECT MAX(Plan\_Claimed) FROM Insurence\_count IC2

WHERE IC1.DiseaseID= IC2.DiseaseID ))) AS t ;

**Output 2.3:**

|  |  |  |  |
| --- | --- | --- | --- |
| diseaseID | planName | plan\_claimed | Plan\_Count |
| 1 | Alpa Bima Group | 1 | 1 |
| 1 | Arogya Raksha- Group Health Insurance Scheme | 1 | 2 |
| 1 | Arogya Sanjeevani | 1 | 3 |
| 1 | Arogya Supreme | 1 | 4 |
| 1 | Care Classic | 1 | 5 |
| 1 | Care Plus | 1 | 6 |
| 1 | Chola Group EMI Protect | 1 | 7 |
| 1 | Chola Group Top Up Protect | 1 | 8 |
| 1 | Digit Group Total Protect Policy | 1 | 9 |

|  |
| --- |
| **Problem Statement 2.4:**  The Healthcare department wants to know which disease is most likely to infect multiple people in the same household. For each disease find the number of households that has more than one patient with the same disease.  Note: 2 people are considered to be in the same household if they have the same address. |

**Solution 2.4 :**

SELECT diseaseid, diseaseName, COUNT (addressID) AS 'Number of Households'

FROM (SELECT diseaseID,diseaseName, addressID

FROM disease

JOIN treatment USING(diseaseID)

JOIN Person on Person.PersonID = treatment.PatientID

GROUP BY diseaseID, AddressID

HAVING COUNT (distinct patientID) > 1

ORDER BY diseaseID) AS q1

GROUP BY diseaseID;

**Output 2.4:**

|  |  |  |
| --- | --- | --- |
| diseaseid | diseaseName | Number of Households |
| 1 | Alzheimer's disease | 7 |
| 2 | Amyotrophic lateral sclerosis | 12 |
| 3 | Anorexia nervosa | 8 |
| 4 | Anxiety disorder | 8 |
| 5 | Asthma | 12 |
| 6 | Atherosclerosis | 12 |
| 7 | Attention deficit hyperactivity disorder | 6 |
| 8 | Autism | 8 |
| 9 | Autoimmune diseases | 7 |

|  |
| --- |
| **Problem Statement 2.5:**  An Insurance company wants a state wise report of the treatments to claim ratio between 1st April 2021 and 31st March 2022 (days both included). Assist them to create such a report. |

**Solution 2.5 :**

SELECT state, COUNT ( distinct treatmentID)/COUNT (distinct claimID) AS 'Treatment to

Claim Ratio'

FROM treatment

JOIN person on person.personID = treatment.patientID

JOIN address USING(addressID)

WHERE date BETWEEN '2021-04-01' AND '2022-03-31'

GROUP BY state

ORDER BY state;

**Output 2.5:**

|  |  |
| --- | --- |
| state | Treatment to Claim Ratio |
| AK | 1.4627 |
| AL | 1.6385 |
| AR | 1.5326 |
| AZ | 1.6463 |
| CA | 1.467 |
| CO | 1.5965 |
| CT | 1.4519 |
| DC | 1.5182 |
| FL | 1.6842 |
| GA | 1.5354 |
| KY | 1.4713 |
| MA | 1.4792 |
| MD | 1.5182 |
| OK | 1.6829 |
| TN | 1.6911 |
| VT | 1.4719 |

Solution 3

Problem Statements

|  |
| --- |
| **Problem Statement 3.1:**  Some complaints have been lodged by patients that they have been prescribed hospital-exclusive medicine that they can’t find elsewhere and facing problems due to that. Joshua, from the pharmacy management, wants to get a report of which pharmacies have prescribed hospital-exclusive medicines the most in the years 2021 and 2022. Assist Joshua to generate the report so that the pharmacies who prescribe hospital-exclusive medicine more often is advised to avoid such practice if possible. |

**Solution 3.1:**

SELECT pharmacyID, pharmacyName, COUNT (distinct prescriptionID) AS

Prescription\_COUNT , COUNT (medicineID) AS Medicine\_COUNT

FROM medicine

JOIN contain USING(medicineID)

JOIN prescription USING(prescriptionID)

JOIN treatment USING(treatmentID)

JOIN pharmacy USING(pharmacyID)

WHERE hospitalExclusive = 'S' AND year(date) in( 2021 , 2022)

GROUP BY pharmacyID

ORDER BY Medicine\_COUNT DESC, prescription\_COUNT DESC;

**Output 3.1:**

|  |  |  |  |
| --- | --- | --- | --- |
| pharmacyID | pharmacyName | Prescription\_Count | Medicine\_Count |
| 9296 | Gem Drugs | 27 | 43 |
| 7448 | Central Drug Store | 29 | 39 |
| 5058 | Bio Scrip | 26 | 37 |
| 6914 | Prescription Lifeline | 22 | 37 |
| 2426 | Lifeshave Pharmacy | 21 | 36 |
| 9681 | Be Well | 23 | 35 |
| 3651 | Good RX | 25 | 34 |
| 3536 | Outpatient Pharmacy | 24 | 34 |
| 5659 | True Pill | 23 | 32 |

|  |
| --- |
| **Problem Statement 3.2:**  Insurance companies want to assess the performance of their insurance plans.  Generate a report that shows each insurance plan, the company that issues the plan, and the number of treatments the plan was claimed for. |

**Solution 3.2:**

SELECT companyID, companyName, planName, COUNT (claimID) AS 'Number\_of\_claims'

FROM treatment

JOIN claim USING (claimID)

JOIN insurancePlan USING (UIN)

JOIN insurancecompany USING(companyID)

GROUP BY companyID, planName

ORDER BY companyID, planName;

**Output 3.2:**

|  |  |  |  |
| --- | --- | --- | --- |
| companyID | companyName | planName | Number\_of\_claims |
| 1118 | Star Health and Allied Insurrance Co. Ltd. | Star Group Health Insurance Policy Gold (For Bank Customers) | 30 |
| 1409 | Niva Bupa Health Insurance Co. Ltd. | Smart Health + (Add-On) | 41 |
| 1409 | Niva Bupa Health Insurance Co. Ltd. | Zero Co-Pay | 28 |
| 1583 | Iffco Tokio General Insurance Company Ltd. | Wellness Addon | 36 |
| 1583 | Iffco Tokio General Insurance Company Ltd. | Welness Benefit Addon(Group) | 32 |
| 1839 | Bajaj Allianz General Insurance Co. Ltd. | Family Care | 49 |
| 1839 | Bajaj Allianz General Insurance Co. Ltd.ï¿½ | M Care (Group) | 41 |
| 1839 | Bajaj Allianz General Insurance Co. Ltd. | M Care | 36 |
| 1933 | HDFC ERGO General Insurance Company Ltd. | Optima Restore | 43 |

|  |
| --- |
| **Problem Statement 3.3 :**  Insurance companies want to assess the performance of their insurance plans.  Generate a report that shows each insurance company's name with their most and least claimed insurance plans. |

**Solution 3.3 :**

SELECT CompanyID, companyName, Minimum\_claimed, Number\_of\_Minimum\_Claims,

Maximum\_claimed, Number\_of\_Maximum\_Claims

FROM ( SELECT companyID, group\_concat(planName separator ', ') AS Minimum\_claimed,

Number\_of\_Claims AS Number\_of\_Minimum\_Claims FROM planInfo

WHERE min\_r = 1

GROUP BY companyID, Number\_of\_Claims ) AS mi

JOIN ( SELECT companyID, group\_concat(planName separator ', ') AS Maximum\_claimed,

Number\_of\_Claims AS Number\_of\_Maximum\_Claims

FROM planInfo

WHERE max\_r = 1

GROUP BY companyID, Number\_of\_Claims) AS ma USING (companyID)

JOIN insurancecompany USING(companyID);

**Output 3.3:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| CompanyID | companyName | Minimum\_claimed | Number\_of\_Minimum\_Claims | Maximum\_claimed | Number\_of\_Maximum\_Claims |
| 1118 | Star Health and Allied Insurrance Co. Ltd. | Star Group Health Insurance Policy Gold (For Bank Customers) | 30 | Star Group Health Insurance Policy Gold (For Bank Customers) | 30 |
| 1409 | Niva Bupa Health Insurance Co. Ltd. | Zero Co-Pay | 28 | Smart Health + (Add-On) | 41 |
| 1583 | Iffco Tokio General Insurance Company Ltd. | Welness Benefit Addon(Group) | 32 | Wellness Addon | 36 |
| 1839 | Bajaj Allianz General Insurance Co. Ltd.ï¿½ | M Care | 36 | Family Care | 49 |
| 1933 | HDFC ERGO General Insurance Company Ltd. | Unlimited Restore (Add On ) | 34 | Optima Restore | 43 |
| 1999 | Cholamandalam MS General | Chola Group Top Up Protect | 23 | Chola Group Top Up Protect | 23 |
| 2295 | Bajaj Allianz General Insuarnce Co. Ltd | Health Prime (Group) | 32 | Silver Health | 43 |
| 2725 | Tata AIG General Insuarnce Co Ltd | Tata AIG Criti- Medicare | 37 | Tata AIG Criti- Medicare | 37 |

|  |
| --- |
| **Problem Statement 3.4 :**  The healthcare department wants a state-wise health report to assess which state requires more attention in the healthcare sector. Generate a report for them that shows the state name, number of registered people in the state,  number of registered patients in the state, and the people-to-patient ratio.  sort the data by people-to-patient ratio. |

**Solution 3.4 :**

SELECT state, COUNT (personID) AS registered\_people, COUNT ( distinct patientID) as

registered\_patients

FROM address

JOIN person AS p USING (addressID)

left JOIN patient AS pa on pa.patientID = p.personID

GROUP BY state

ORDER BY state;

**Output 3.4:**

|  |  |  |
| --- | --- | --- |
| state | registered\_people | registered\_patients |
| AK | 144 | 52 |
| AL | 182 | 79 |
| AR | 159 | 67 |
| AZ | 165 | 63 |
| CA | 254 | 115 |
| CO | 185 | 76 |
| CT | 173 | 66 |
| DC | 181 | 76 |
| FL | 170 | 75 |
| GA | 173 | 73 |
| KY | 140 | 57 |
| MA | 155 | 57 |
| MD | 142 | 60 |
| OK | 140 | 65 |
| TN | 149 | 79 |
| VT | 166 | 66 |

|  |
| --- |
| **Problem Statement 3.5 :**  Jhonny, from the finance department of Arizona(AZ), has requested a report that lists the total quantity of medicine each pharmacy in his state has prescribed that falls under Tax criteria I for treatments that took place in 2021. Assist Jhonny in generating the report. |

**Solution 3.5 :**

SELECT pharmacyID, pharmacyName, sum(quantity) AS 'Total Medicine'

FROM medicine

JOIN contain USING(medicineID)

JOIN prescription USING(prescriptionID)

JOIN pharmacy USING(pharmacyID)

JOIN address USING(addressID)

JOIN treatment USING(treatmentID)

WHERE taxCriteria = 'I' AND state = 'AZ' AND year(date)= 2021

GROUP BY pharmacyID

ORDER BY pharmacyName;

**Output 3.5:**

|  |  |  |
| --- | --- | --- |
| pharmacyID | pharmacyName | Total Medicine |
| 9681 | Be Well | 364 |
| 1624 | Caremark | 369 |
| 8933 | Cashway Pharmacy | 123 |
| 9659 | Express Scripts | 329 |
| 2218 | Heallergy | 290 |
| 2301 | HealthDirect | 535 |
| 1628 | IDL Drug Stores | 524 |
| 5480 | Kerr Drug | 460 |
| 3104 | Louis And Clark Drug | 348 |
| 8829 | Lyfe Pharmacy | 412 |
| 8897 | MedSavvy | 179 |
| 3799 | Newday Drug Store | 211 |
| 3536 | Outpatient Pharmacy | 567 |
| 1478 | Pocketpills | 411 |
| 5450 | Reliable Rexall | 358 |
| 4938 | University Pharmacy | 448 |
| 8442 | Wellmans Pharmacy | 567 |

Solution 4

SQL CASE Expression

|  |
| --- |
| **Problem Statement 4.1 :**  “HealthDirect” pharmacy finds it difficult to deal with the product type of medicine being displayed in numerical form, they want the product type in words. Also, they want to filter the medicines based on tax criteria. Display only the medicines of product categories 1, 2, and 3 for medicines that come under tax category I and medicines of product categories 4, 5, and 6 for medicines that come under tax category II. Write a SQL query to solve this problem.  ProductType numerical form and ProductType in words are given by  1 - Generic,  2 - Patent,  3 - Reference,  4 - Similar,  5 - New,  6 - Specific,  7 - Biological,  8 – Dinamized |

**Solution 4.1 :**

SELECT distinct medicineID, companyName, productName, description, substanceName,

CASE productType

WHEN 1 THEN 'Generic'

WHEN 2 THEN 'Patent'

WHEN 3 THEN 'Reference'

WHEN 4 THEN 'Similar'

WHEN 5 THEN 'New'

WHEN 6 THEN 'Specific'

END AS producttype, taxCriteria, hospitalexclusive, governmentdiscount, taxImunity, maxPrice

FROM medicine

WHERE ((productType in (1,2,3) AND taxCriteria = 'I') or (productType in (4,5,6) AND taxCriteria = 'II'))

ORDER BY medicineID;

**Output 4.1:**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| medicineID | companyName | productName | description | substanceName | producttype | taxCriteria | hospitalexclusive | governmentdiscount | taxImunity | maxPrice |
| 3 | LIBRA COMERCIO DE PRODUTOS FARMACEUTICOS LTDA | TEICOPLANINA | 400 MG PO LIOFILIZADO SOL INJ CX C/ 1 FR/AMP X 3 M | NC/NI | Generic | I | N | N | N | 296.38 |
| 5 | LIBRA COMERCIO DE PRODUTOS FARMACEUTICOS LTDA | OXALIPLATINA | 50 MG PO LIOFILIZADO FR/AMP X 500 MG | NC/NI | Generic | I | N | N | N | 1186.81 |
| 11 | LIBRA COMERCIO DE PRODUTOS FARMACEUTICOS LTDA | TEICOPLANINA | 200 MG PO LIOFILIZADO SOL INJ CX C/ 1 FR/AMP X 3 M | NC/NI | Generic | I | N | N | N | 148.46 |
| 12 | LIBRA COMERCIO DE PRODUTOS FARMACEUTICOS LTDA | OXALIPLATINA | 100 MG PO LIOFILIZADO FR/AMP X 1000 MG | NC/NI | Generic | I | N | N | N | 2373.63 |
| 13 | LIBRA COMERCIO DE PRODUTOS FARMACEUTICOS LTDA | SULBACTAM SODICO + AMPICILINA SODICA | 1 G + 2 G CT FR AMP VD INC | NC/NI | Similar | II | N | N | N | 29.59 |
| 14 | LIBRA COMERCIO DE PRODUTOS FARMACEUTICOS LTDA | PACLITAXEL | 6 MG/ML SOL INJ CT FR/AMP X 50 ML | NC/NI | Generic | I | N | N | N | 4122.12 |
| 15 | LIBRA COMERCIO DE PRODUTOS FARMACEUTICOS LTDA | PACLITAXEL | 6 MG/ML SOL INJ CX C/ 1 FR/AMP X 17 ML | NC/NI | Generic | I | N | N | N | 1388.75 |
| 26 | SANOFI MEDLEY FARMACEUTICA LTDA. | AMPLOTAL | 1000 MG CX C/FA + DIL | ampicilina sodica|ampicilina benzatina | Similar | II | N | N | N | 7.93 |
| 74 | LABORATORIO SANOBIOL LIMITADA | SOLUCAO FISIOLOGICA DE RINGER COM LACTATO DE SODIO | SOL INJ CX 20 FR 500 ML | lactato de sodio|cloreto de sodio|cloreto de potassio|cloreto de calcio | Specific | II | N | N | N | 2.39 |

|  |
| --- |
| **Problem Statement 4.2 :**  'Ally Scripts' pharmacy company wants to find out the quantity of medicine prescribed in each of its prescriptions. Write a query that finds the sum of the quantity of all the medicines in a prescription and if the total quantity of medicine is less than 20 tag it AS “low quantity”. If the quantity of medicine is FROM 20 to 49 (both numbers including) tag it as “medium quantity“ and if the quantity is more than equal to 50 then tag it AS “high quantity”. Show the prescription Id, the Total Quantity of all the medicines in that prescription, and the Quantity tag for all the prescriptions issued by 'Ally Scripts'. |

**Solution 4.2 :**

SELECT prescriptionID, sum(quantity) AS Total\_quantity,

CASE

WHEN sum(quantity) < 20 THEN 'Low Quantity'

WHEN sum(quantity) BETWEEN 20 AND 49 THEN 'Medium Quantity'

WHEN sum(quantity) > 50 THEN 'High Quantity'

END AS 'Tag'

FROM prescription

JOIN contain USING(prescriptionID)

WHERE pharmacyID = (SELECT pharmacyID

FROM pharmacy

WHERE pharmacyName = 'Ally Scripts')

GROUP BY prescriptionID;

**Output 4.2:**

|  |  |  |  |
| --- | --- | --- | --- |
| prescriptionID | Total\_quantity | Tag |  |
| 1147561399 | 43 | Medium Quantity | |
| 1222719376 | 71 | High Quantity | |
| 1408276190 | 48 | Medium Quantity | |
| 1571383871 | 22 | Medium Quantity | |
| 1668180798 | 52 | High Quantity | |
| 1767949601 | 7 | Low Quantity | |
| 1838926888 | 47 | Medium Quantity | |
| 1840130817 | 66 | High Quantity | |
| 1895387326 | 41 | Medium Quantity | |

|  |
| --- |
| **Problem Statement 4.3 :**  In the Inventory of a pharmacy 'Spot Rx' the quantity of medicine is considered ‘HIGH QUANTITY’ WHEN the quantity exceeds 7500 AND ‘LOW QUANTITY’ WHEN the quantity falls short of 1000. The discount is considered “HIGH” if the discount rate on a product is 30% or higher, AND the discount is considered “NONE” WHEN the discount rate on a product is 0%. 'Spot Rx' needs to find all the Low quantity products with high discount s AND all the high-quantity products with no discount so they can adjust the discount rate according to the demand. Write a query for the pharmacy listing all the necessary details relevant to the given requirement.  Hint: Inventory is reflected in the Keep table. |

**Solution 4.3 :**

SELECT MedicineID, Quantity, DisCOUNT ,

CASE

WHEN (quantity > 7500 AND disCOUNT = 0) THEN 'High quantity products

with no disCOUNT '

WHEN (quantity < 1000 AND disCOUNT >= 30) THEN 'Low quantity products

with high disCOUNT s'

END AS Medicine\_Type

FROM keep

WHERE pharmacyID = (SELECT pharmacyID

FROM pharmacy

WHERE pharmacyName = 'Spot Rx')

AND((quantity > 7500 AND disCOUNT = 0) or (quantity < 1000 AND disCOUNT >= 30))

ORDER BY medicine\_type, quantity;

**Output 4.3:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| MedicineID | Quantity | Discount | Medicine\_Type |  |  |
| 17172 | 7504 | 0 | High quantity products with no discount | | |
| 41404 | 7560 | 0 | High quantity products with no discount | | |
| 53209 | 7618 | 0 | High quantity products with no discount | | |
| 39816 | 7664 | 0 | High quantity products with no discount | | |
| 19571 | 7756 | 0 | High quantity products with no discount | | |
| 15999 | 7790 | 0 | High quantity products with no discount | | |
| 26749 | 7835 | 0 | High quantity products with no discount | | |
| 35997 | 7853 | 0 | High quantity products with no discount | | |
| 50031 | 8094 | 0 | High quantity products with no discount | | |

|  |
| --- |
| **Problem Statement 4.4 :**  Mack, from HealthDirect Pharmacy, wants to get a list of all the affordable AND costly, hospital-exclusive medicines in the database. WHERE affordable medicines are the medicines that have a maximum price of less than 50% of the avg maximum price of all the medicines in the database, AND costly medicines are the medicines that have a maximum price of more than double the avg maximum price of all the medicines in the database. Mack wants clear text next to each medicine name to be displayed that identifies the medicine AS affordable or costly. The medicines that do not fall under either of the two categories need not be displayed. Write a SQL query for Mack for this requirement. |

**Solution 4.4 :**

set @avg\_price = (SELECT avg(maxPrice) FROM medicine);

SELECT medicineID, companyName, productName,

CASE

WHEN maxPrice < @avg\_price\*0.5 THEN 'Affordable'

WHEN maxPrice > @avg\_price\*2 THEN 'Costly'

END AS Tag

FROM medicine AS m

WHERE hospitalExclusive = 's'

AND ((maxPrice < @avg\_price\*0.5) OR (maxPrice > @avg\_price\*2))

ORDER BY medicineID;

**Output 4.4:**

|  |  |  |  |
| --- | --- | --- | --- |
| medicineID | companyName | productName | Tag |
| 135 | PHARLAB INDUSTRIA FARMACEUTICA S.A. | BISOLPHAR | Affordable |
| 140 | CRISTALIA PRODUTOS QUIMICOS FARMACEUTICOS LTDA. | FUMARATO DE QUETIAPINA | Costly |
| 142 | EUROFARMA LABORATORIOS S.A. | OXACILINA SODICA | Affordable |
| 149 | CRISTALIA PRODUTOS QUIMICOS FARMACEUTICOS LTDA. | OMEPRAZOL SODICO | Costly |
| 152 | EUROFARMA LABORATORIOS S.A. | SOLUCAO FISIOLOGICA DE RINGER | Affordable |
| 156 | EVOLABIS PRODUTOS FARMACEUTICOS LTDA | EVOLOX | Affordable |
| 187 | DR. REDDYS FARMACEUTICA DO BRASIL LTDA | PACLIRED | Costly |
| 188 | EMS S/A | PIROXICAM | Affordable |
| 199 | ACCORD FARMACEUTICA LTDA | MICOFENOLATO DE MOFETILA | Costly |

|  |
| --- |
| **Problem Statement 4.5 :**  The healthcare department wants to categorize the patients into the following category.  YoungMale: Born on or after 1st Jan 2005 and gender male.  YoungFemale: Born on or after 1st Jan 2005 and gender female.  AdultMale: Born before 1st Jan 2005 but on or after 1st Jan 1985 and gender male.  AdultFemale: Born before 1st Jan 2005 but on or after 1st Jan 1985 and gender female.  MidAgeMale: Born before 1st Jan 1985 but on or after 1st Jan 1970 and gender male.  MidAgeFemale: Born before 1st Jan 1985 but on or after 1st Jan 1970 and gender female.  ElderMale: Born before 1st Jan 1970, and gender male.  ElderFemale: Born before 1st Jan 1970, and gender female.  Write a SQL query to list all the patient name, gender, dob, and their category. |

**Solution 4.5 :**

CREATE FUNCTION category(dob date, gender varchar(10))

RETURNS VARCHAR(50)

deterministic

BEGIN

DECLARE category varchar(50) default NULL;

if dob > '2005-01-01' AND gender = 'Male' THEN set category = 'YoungMale' ;

elseif dob > '2005-01-01' AND gender = 'Female' THEN set category =

'YoungFemale';

elseif dob > '1985-01-01' AND gender = 'Male' THEN set category =

'AdultMale';

elseif dob > '1985-01-01' AND gender = 'Female' THEN set category =

'AdultFemale';

elseif dob > '1970-01-01' AND gender = 'Male' THEN set category =

'MidAgeMale';

elseif dob > '1970-01-01' AND gender = 'Female' THEN set category =

'MidAgeFemale';

elseif dob < '1970-01-01' AND gender = 'Male' THEN set category =

'ElderMale';

elseif dob < '1970-01-01' AND gender = 'Female' THEN set category =

'ElderFemale';

END if;

return (category);

END $$

delimiter ;

SELECT personName, gender, dob, category(dob, gender) AS 'Category'

FROM patient

JOIN person on person.personId = patient.patientID;

**Output 4.5:**

|  |  |  |  |
| --- | --- | --- | --- |
| personName | gender | dob | Category |
| Cathy Bocook | female | 02-09-2008 | YoungFemale |
| Joseph Foglesong | male | 11-04-1965 | ElderMale |
| Phillip Mackey | male | 01-12-1963 | ElderMale |
| Douglas Robertson | male | 02-10-2017 | YoungMale |
| Mike Pray | male | 11-04-1956 | ElderMale |
| Joe Brice | male | 21-04-1954 | ElderMale |
| Thomas Boyle | male | 11-02-1965 | ElderMale |
| Herman Buchner | male | 06-06-1965 | ElderMale |
| Anthony Reagan | male | 28-03-1954 | ElderMale |
| Lucretia Holmes | female | 20-01-1962 | ElderFemale |
| Stephen Bergeron | male | 16-06-1959 | ElderMale |
| Natasha Harrison | female | 06-04-1949 | ElderFemale |
| Clarence Lee | male | 29-03-2022 | YoungMale |
| Essie Moroz | female | 05-07-1986 | AdultFemale |
| Elvira Cummins | female | 02-11-1936 | ElderFemale |
| Madge Hancock | female | 05-04-2019 | YoungFemale |
| Raymond Blakey | male | 26-10-1951 | ElderMale |
| Jason Riecke | male | 01-11-2021 | YoungMale |
| Kelly Gill | female | 12-09-1935 | ElderFemale |

Solution 5

SQL Grouping

|  |
| --- |
| **Problem Statement 5.1 :**  Johansson is trying to prepare a report on patients who have gone through treatments more than once. Help Johansson prepare a report that shows the patient's name, the number of treatments they have undergone, and their age, Sort the data in a way that the patients who have undergone more treatments appear on top. |

**Solution 5.1 :**

SELECT personName, COUNT (treatmentID) AS 'Number\_of\_Treatments', age

FROM person

JOIN patient on patient.patientID = person.personID

JOIN treatment USING(patientID)

GROUP BY patientID

HAVING COUNT (treatmentID) > 1

ORDER BY Number\_of\_Treatments desc;

**Output 5.1:**

|  |  |  |
| --- | --- | --- |
| personName | Number\_of\_Treatments | age |
| Allen Brown | 43 | 39 |
| Mary Nelson | 43 | 61 |
| Gregory Hansen | 42 | 0 |
| James James | 42 | 34 |
| Herman Buchner | 42 | 56 |
| Stephanie Weaver | 42 | 9 |
| Billy Rave | 42 | 69 |
| Michael Runyon | 41 | 63 |
| Alex Smith | 41 | 47 |

|  |
| --- |
| **Problem Statement 5.2 :**  Bharat is researching the impact of gender on different diseases, He wants to analyze if a certain disease is more likely to infect a certain gender or not. Help Bharat analyze this by creating a report showing for every disease how many males and females underwent treatment for each in the year 2021.  It would also be helpful for Bharat if the male-to-female ratio is also shown. |

**Solution 5.2 :**

With patient\_data AS

(SELECT \*

FROM person

JOIN patient on patient.patientID = person.personID

JOIN treatment USING(patientID)

WHERE year(date)=2021)

SELECT diseaseID, DiseaseName, Male\_COUNT , Female\_COUNT ,

Male\_COUNT /Female\_COUNT AS 'Male\_to\_Female\_ratio'

FROM (SELECT diseaseID, COUNT (treatmentID) AS 'Male\_COUNT '

FROM Patient\_data

WHERE gender = 'male'

GROUP BY diseaseID) AS m

JOIN (SELECT diseaseID, COUNT (treatmentID) AS 'Female\_COUNT '

FROM Patient\_data

WHERE gender = 'Female'

GROUP BY diseaseID) AS f USING (diseaseID)

JOIN disease USING(diseaseID)

ORDER BY diseaseID;

**Output 5.2:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| diseaseID | DiseaseName | Male\_Count | Female\_Count | Male\_to\_Female\_ratio |
| 1 | Alzheimer's disease | 45 | 23 | 1.9565 |
| 2 | Amyotrophic lateral sclerosis | 32 | 25 | 1.28 |
| 3 | Anorexia nervosa | 40 | 28 | 1.4286 |
| 4 | Anxiety disorder | 33 | 25 | 1.32 |
| 5 | Asthma | 41 | 25 | 1.64 |
| 6 | Atherosclerosis | 50 | 28 | 1.7857 |
| 7 | Attention deficit hyperactivity disorder | 43 | 37 | 1.1622 |
| 8 | Autism | 40 | 24 | 1.6667 |
| 9 | Autoimmune diseases | 33 | 19 | 1.7368 |

|  |
| --- |
| **Problem Statement 5.3 :**  Kelly, from the Fortis Hospital management, has requested a report that shows for each disease, the top 3 cities that had the most number treatment for that disease. Generate a report for Kelly’s requirement. |

**Solution 5.3 :**

SELECT diseaseID, DiseaseName, City, Number\_of\_Patients FROM

(SELECT diseaseID, city,COUNT (treatmentID) AS 'Number\_of\_Patients',rank()

over(partition by diseaseID ORDER BY COUNT (treatmentID) desc) AS

'City\_rank'

FROM treatment

JOIN person on person.personID = treatment.patientID

JOIN address USING(addressID)

GROUP BY diseaseID, city

ORDER BY diseaseid) AS q1

JOIN disease USING(diseaseID)

WHERE city\_rank<=3

ORDER BY diseaseID, Number\_of\_patients desc;

**Output 5.3:**

|  |  |  |  |
| --- | --- | --- | --- |
| diseaseID | DiseaseName | City | Number\_of\_Patients |
| 1 | Alzheimer's disease | Washington | 28 |
| 1 | Alzheimer's disease | Montgomery | 20 |
| 1 | Alzheimer's disease | Savannah | 16 |
| 2 | Amyotrophic lateral sclerosis | Montgomery | 22 |
| 2 | Amyotrophic lateral sclerosis | Washington | 21 |
| 2 | Amyotrophic lateral sclerosis | Oklahoma City | 18 |
| 3 | Anorexia nervosa | Montgomery | 25 |
| 3 | Anorexia nervosa | Nashville | 22 |
| 3 | Anorexia nervosa | Manchester | 21 |

|  |
| --- |
| **Problem Statement 5.4 :**  Brooke is trying to figure out if patients with a particular disease are preferring some pharmacies over others or not, For this purpose, she has requested a detailed pharmacy report that shows each pharmacy name, AND how many prescriptions they have prescribed for each disease in 2021 AND 2022, She expects the number of prescriptions prescribed in 2021 AND 2022 be displayed in two separate columns. Write a query for Brooke’s requirement. |

**Solution 5.4 :**

with pres as

(SELECT pharmacyID, diseaseID,date, prescriptionID

FROM prescription

JOIN treatment USING(treatmentID))

SELECT pharmacyID, pharmacyName, diseaseID, diseaseName,

coalesce(prescription\_COUNT \_2021, 0) AS 'prescription\_COUNT \_2021',

coalesce(prescription\_COUNT \_2022, 0) AS 'prescription\_COUNT \_2022'

FROM pharmacy

JOIN disease

left JOIN (SELECT pharmacyID, diseaseID, COUNT (prescriptionID) AS

prescription\_COUNT \_2021

FROM pres

WHERE year(DATE) = 2021

GROUP BY pharmacyID, diseaseID) AS Y2K1 USING(pharmacyID, diseaseID)

left JOIN (SELECT pharmacyID, diseaseID, COUNT (prescriptionID) AS

prescription\_COUNT \_2022

FROM pres

WHERE year(DATE) = 2022

GROUP BY pharmacyID, diseaseID) AS Y2K2 USING(pharmacyID, diseaseID)

WHERE prescription\_COUNT \_2021 !=0 AND prescription\_COUNT \_2022 != 0

ORDER BY pharmacyID, prescription\_COUNT \_2021 desc , prescription\_COUNT \_2022 desc;

**Output 5.4:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| pharmacyID | Pharmacy  Name | diseaseID | diseaseName | | prescription\_count\_2021 | prescription\_count\_2022 |
| 1008 | MobiMeds | 20 | Dilated cardiomyopathy | 2 | | 1 |
| 1008 | MobiMeds | 28 | Myocardial infarction | 2 | | 1 |
| 1008 | MobiMeds | 12 | Chronic fatigue syndrome | 1 | | 1 |
| 1008 | MobiMeds | 19 | Diabetes mellitus type 2 | 1 | | 1 |
| 1008 | MobiMeds | 22 | Guillain syndrome | 1 | | 1 |
| 1008 | MobiMeds | 23 | Irritable bowel syndrome | 1 | | 1 |
| 1008 | MobiMeds | 29 | Obesity | 1 | | 1 |
| 1145 | Spot Rx | 29 | Obesity | 2 | | 2 |
| 1145 | Spot Rx | 9 | Autoimmune diseases | 2 | | 1 |

|  |
| --- |
| **Problem Statement 5.5 :**  Walde, FROM Rock tower insurance, has sent a requirement for a report that presents which insurance company is targeting the patients of which state the most. Write a query for Walde that fulfills the requirement of Walde. Note: We can assume that the insurance company is targeting a region more if the patients of that region are claiming more insurance of that company. |

**Solution 5.5 :**

with state\_pref as

(SELECT companyID, state, COUNT (claimID) AS 'patient\_COUNT '

FROM address

JOIN person USING(addressID)

JOIN treatment on person.personID = treatment.patientID

JOIN claim USING(claimID)

JOIN insuranceplan USING(UIN)

GROUP BY companyID, state

ORDER BY companyID, COUNT (claimID) desc)

SELECT companyID, companyName, COUNT (\*) AS 'Number\_of\_states',

group\_concat(state separator ', ') AS states, patient\_COUNT

FROM state\_pref AS s1 JOIN insurancecompany USING(companyID)

WHERE patient\_COUNT = (SELECT max(patient\_COUNT )

FROM state\_pref AS s2 WHERE s1.companyID = s2.companyID)

GROUP BY companyID, patient\_COUNT

ORDER BY companyID;

**Output 5.5:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| companyID | companyName | Number\_of\_states | states | patient\_count |
| 1118 | Star Health and Allied Insurrance Co. Ltd. | 1 | OK | 5 |
| 1409 | Niva Bupa Health Insurance Co. Ltd. | 3 | AK, CA, OK | 8 |
| 1583 | Iffco Tokio General Insurance Company Ltd. | 1 | CT | 8 |
| 1839 | Bajaj Allianz General Insurance Co. Ltd.ï¿½ | 2 | CA, CO | 13 |
| 1933 | HDFC ERGO General Insurance Company Ltd. | 1 | CA | 11 |
| 1999 | Cholamandalam MS General | 1 | AL | 5 |
| 2295 | Bajaj Allianz General Insuarnce Co. Ltd | 1 | CA | 22 |

Solution 6

SQL Grouping 2

|  |
| --- |
| **Problem Statement 6.1 :**  The healthcare department wants a pharmacy report on the percentage of hospital-exclusive medicine prescribed in the year 2022. Assist the healthcare department to view for each pharmacy, the pharmacy id, pharmacy name, total quantity of medicine prescribed in 2022, total quantity of hospital-exclusive medicine prescribed by the pharmacy in 2022, AND the percentage of hospital-exclusive medicine to the total medicine prescribed in 2022. Order the result in descending order of the percentage found. |

**Solution 6.1 :**

with pharmacy\_COUNT as

(SELECT pharmacyID, quantity, hospitalExclusive

FROM treatment

JOIN prescription USING(treatmentID)

JOIN contain USING(prescriptionID)

JOIN medicine USING(medicineID)

WHERE year(date) = 2022)

SELECT pharmacyID, pharmacyName, Total\_Quantity, Hospital\_Exclusive\_Quantity,

(Hospital\_Exclusive\_Quantity\*100/Total\_Quantity) AS 'Percentage\_Hospital\_Exclusive'

FROM (SELECT pharmacyID, sum(quantity) AS 'Total\_Quantity'

FROM pharmacy\_COUNT

GROUP BY pharmacyID) AS t1

JOIN (SELECT pharmacyID, sum(quantity) AS 'Hospital\_Exclusive\_Quantity'

FROM pharmacy\_COUNT

WHERE hospitalexclusive = 's'

GROUP BY pharmacyID) AS t2 USING(pharmacyID)

JOIN pharmacy USING (pharmacyID)

ORDER BY Percentage\_Hospital\_Exclusive desc;

**Output 6.1:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| pharmacyID | Pharmacy  Name | Total\_Quantity | Hospital\_Exclusive\_Quantity | Percentage\_Hospital\_Exclusive |
| 5246 | Better Life | 159 | 50 | 31.4465 |
| 9999 | Planet Health | 547 | 167 | 30.5302 |
| 6914 | Prescription Lifeline | 796 | 231 | 29.0201 |
| 5866 | Carepoint | 931 | 267 | 28.6788 |
| 8549 | Cash Saver Pharmacy | 763 | 210 | 27.5229 |
| 4025 | Keystone Pharmacy | 553 | 152 | 27.4864 |
| 7532 | Sure Save | 561 | 151 | 26.9162 |
| 5200 | Heartland Pharmacy | 766 | 203 | 26.5013 |
| 2426 | Lifeshave Pharmacy | 1136 | 289 | 25.4401 |
| 5953 | Brennen Drugs Co | 383 | 96 | 25.0653 |
| 9675 | Customceutical Compounding | 525 | 131 | 24.9524 |
| 9010 | Pavilions Pharmacy | 335 | 82 | 24.4776 |
| 9647 | Covenant Pharmacy | 803 | 195 | 24.2839 |
| 3703 | Bioplus Specialty | 445 | 108 | 24.2697 |
| 6971 | Hearty Mart | 541 | 131 | 24.2144 |
| 6674 | The Downtown Dispensary | 415 | 100 | 24.0964 |
| 1624 | Caremark | 725 | 171 | 23.5862 |
| 6444 | Postmeds | 883 | 208 | 23.5561 |
| 7357 | MedShoppe | 962 | 225 | 23.3888 |

|  |
| --- |
| **Problem Statement 6.2 :**  Sarah, FROM the healthcare department, has noticed many people do not claim insurance for their treatment. She has requested a state-wise report of the percentage of treatments that took place without claiming insurance. Assist Sarah by creating a report as per her requirement. |

**Solution 6.2 :**

SELECT state, COUNT (treatmentID) AS 'Total\_Treatments', COUNT (distinct claimID)

AS 'Total\_Claimed', round((1-COUNT (distinct claimID)/COUNT (treatmentID))\*100,2)

AS 'Percentage\_Not\_Claimed'

FROM treatment

JOIN person on person.personID = treatment.patientID

JOIN address USING(addressID)

GROUP BY state;

**Output 6.2:**

|  |  |  |  |
| --- | --- | --- | --- |
| state | Total\_Treatments | Total\_Claimed | Percentage\_Not\_Claimed |
| AK | 428 | 278 | 35.05 |
| AL | 828 | 548 | 33.82 |
| AR | 591 | 375 | 36.55 |
| AZ | 570 | 358 | 37.19 |
| CA | 1092 | 729 | 33.24 |
| CO | 718 | 465 | 35.24 |
| CT | 698 | 442 | 36.68 |
| DC | 719 | 476 | 33.8 |
| FL | 741 | 460 | 37.92 |
| GA | 707 | 451 | 36.21 |
| KY | 469 | 300 | 36.03 |
| MA | 529 | 346 | 34.59 |
| MD | 630 | 410 | 34.92 |
| OK | 788 | 474 | 39.85 |
| TN | 790 | 483 | 38.86 |
| VT | 587 | 368 | 37.31 |

|  |
| --- |
| **Problem Statement 6.3:**  Sarah, FROM the healthcare department, is trying to understand if some diseases are spreading in a particular region. Assist Sarah by creating a report which shows for each state, the number of the most and least treated diseases by the patients of that state in the year 2022. |

**Solution 6.3 :**

with patient\_COUNT as

(SELECT state, DiseaseName, COUNT (patientID) AS 'Number\_of\_Patients',

rank() over(partition by state ORDER BY COUNT (patientID)) AS min\_r,

rank() over(partition by state ORDER BY COUNT (patientID) desc) AS max\_r

FROM disease JOIN treatment USING(diseaseID)

JOIN person on person.personId = treatment.patientID

JOIN address USING(addressID)

GROUP BY state, diseaseID ORDER BY state)

SELECT state, Leaset\_Treated\_Diseases, Number\_of\_Least\_Patients,

Most\_Treated\_Diseases, Number\_of\_Most\_Patients

FROM (SELECT state, group\_concat(diseaseName separator ', ') AS

Leaset\_Treated\_Diseases, Number\_of\_Patients AS Number\_of\_Least\_Patients

FROM patient\_COUNT WHERE min\_r = 1

GROUP BY state, Number\_of\_Patients) AS mi

JOIN (SELECT state, group\_concat(diseaseName separator ', ') AS Most\_Treated\_Diseases,

Number\_of\_Patients AS Number\_of\_Most\_Patients

FROM patient\_COUNT WHERE max\_r = 1

GROUP BY state, Number\_of\_Patients) AS ma USING(state);

**Output 6.3:**

|  |  |  |  |
| --- | --- | --- | --- |
| state | Total\_Treatments | Total\_Claimed | Percentage\_Not\_Claimed |
| AK | 428 | 278 | 35.05 |
| AL | 828 | 548 | 33.82 |
| AR | 591 | 375 | 36.55 |
| AZ | 570 | 358 | 37.19 |
| CA | 1092 | 729 | 33.24 |
| CO | 718 | 465 | 35.24 |
| CT | 698 | 442 | 36.68 |
| DC | 719 | 476 | 33.8 |
| FL | 741 | 460 | 37.92 |
| GA | 707 | 451 | 36.21 |
| KY | 469 | 300 | 36.03 |
| MA | 529 | 346 | 34.59 |
| MD | 630 | 410 | 34.92 |
| OK | 788 | 474 | 39.85 |
| TN | 790 | 483 | 38.86 |
| VT | 587 | 368 | 37.31 |

|  |
| --- |
| **Problem Statement 6.4 :**  Manish, FROM the healthcare department, wants to know how many registered people are registered AS patients AS well, in each city. Generate a report that shows each city that has 10 or more registered people belonging to it AND the number of patients FROM that city AS well AS the percentage of the patient with respect to the registered people. |

**Solution 6.4 :**

SELECT city, COUNT (distinct personID) AS 'Total\_Persons', COUNT (distinct patientID)

AS 'Total\_Patient', round((1-COUNT (distinct patientID)/COUNT (personID))\*100, 2) AS 'Percentage\_Patient\_to\_Person'

FROM address

JOIN person USING(addressID)

LEFT JOIN patient on person.personID = patient.patientID

GROUP BY city

HAVING COUNT (personID) >= 10;

**Output 6.4:**

|  |  |  |  |
| --- | --- | --- | --- |
| city | Total\_Persons | Total\_Patient | Percentage\_Patient\_to\_Person |
| Anchorage | 135 | 50 | 62.96 |
| Annapolis | 35 | 15 | 57.14 |
| Arvada | 144 | 61 | 57.64 |
| Burlington | 10 | 2 | 80 |
| Calhoun | 10 | 7 | 30 |
| Castro Valley | 10 | 3 | 70 |
| Edmond | 18 | 11 | 38.89 |
| Fayetteville | 149 | 64 | 57.05 |
| Fremont | 26 | 15 | 42.31 |
| Glen Burnie | 23 | 10 | 56.52 |
| Glendale | 153 | 60 | 60.78 |
| Hayward | 21 | 10 | 52.38 |
| Livermore | 12 | 5 | 58.33 |
| Louisville | 131 | 53 | 59.54 |

|  |
| --- |
| **Problem Statement 6.5 :**  It is suspected by healthcare research department that the substance “ranitidine” might be causing some side effects. Find the top 3 companies USING the substance in their medicine so that they can be informed about it. |

**Solution 6.5 :**

SELECT COUNT (medicineid) medicine\_count ,companyname

FROM medicine

WHERE substanceName rlike 'ranitidina'

GROUP BY companyname

ORDER BY c desc

LIMIT 3;

**Output 6.5:**

|  |  |
| --- | --- |
| medicine\_count | companyname |
| 39 | GEOLAB INDUSTRIA FARMACEUTICA S/A |
| 16 | LABORATORIO TEUTO BRASILEIRO S/A |
| 13 | EMS S/A |

Solution 7

SQL IF-THEN-ELSE

|  |
| --- |
| **Problem Statement 7.1 :**  Insurance companies want to know if a disease is claimed higher or lower than average. Write a stored procedure that returns “claimed higher than average”  or “claimed lower than average” when the diseaseID is passed to it. Hint: Find average number of insurance claims for all the diseases. If the number of claims for the passed disease is higher than the average return “claimed higher than average” otherwise “claimed lower than average”. |

**Solution 7.1 :**

DELIMITER //

Create procedure Claim\_type(IN disease\_ID\_IN int)

BEGIN

if (SELECT COUNT (distinct claimID) FROM treatment

WHERE diseaseID = disease\_ID\_IN) > (SELECT COUNT (distinct claimID)/COUNT

(distinct diseaseID) FROM treatment)

THEN SELECT disease\_ID\_IN,'Claim Higher than average' AS Claim\_type;

ELSE SELECT disease\_ID\_IN, 'Claim Lower than average' AS Claim\_type;

END if;

END //

DELIMITER ;

call claim\_type(15);

**Output 7.1:**

|  |  |
| --- | --- |
| disease\_ID\_IN | Claim\_type |
| 15 | Claim Lower than average |

|  |
| --- |
| **Problem Statement 7.2 :**  Joseph FROM Healthcare department has requested for an application which helps him get genderwise report for any disease. Write a stored procedure when passed a disease\_id returns 4 columns, disease\_name, number\_of\_male\_treated, number\_of\_female\_treated, more\_treated\_gender Where, more\_treated\_gender is either ‘male’ or ‘female’ based on which gender underwent more often for the disease, if the number is same for both the genders, the value should be ‘same’. |

**Solution 7.2 :**

DELIMITER //

Create procedure Disease\_Detail(IN disease\_ID\_IN int)

BEGIN

SELECT DiseaseName, Male\_COUNT , Female\_COUNT , if( male\_COUNT >

Female\_COUNT , 'Male', if (female\_COUNT > male\_COUNT , 'Female', 'Same')) AS 'More\_Treated\_Gender'

FROM (SELECT COUNT (gender) AS Male\_COUNT

FROM treatment JOIN person on person.personID = treatment.patientID WHERE diseaseID = disease\_ID\_IN AND gender = 'male') AS m

JOIN (SELECT COUNT(gender) AS Female\_COUNT

FROM treatment JOIN person on person.personID = treatment.patientID WHERE diseaseID = disease\_ID\_IN AND gender = 'Female') AS f

JOIN (SELECT diseaseName FROM disease WHERE diseaseID = Disease\_ID\_IN) AS d ;

END //

DELIMITER ;

CALL Disease\_Detail(10);

**Output 7.2:**

|  |  |  |  |
| --- | --- | --- | --- |
| diseaseName | Male\_count | Female\_Count | More\_Treated\_Gender |
| Bipolar disorder | 166 | 114 | Male |

|  |
| --- |
| **Problem Statement 7.3 :**  The insurance companies want a report on the claims of different insurance plans. Write a query that finds the top 3 most and top 3 least claimed insurance plans.The query is expected to return the insurance plan name, the insurance company name which has that plan, and whether the plan is the most claimed or least claimed. |

**Solution 7.3 :**

with Claim\_COUNT AS

(SELECT CompanyID, PlanName, COUNT (claimID) AS Claim\_COUNT s,

rank() over(ORDER BY COUNT (claimID)) AS 'Min\_r',

rank() over(ORDER BY COUNT (claimID) desc) AS 'Max\_r'

FROM insuranceplan

JOIN claim USING(UIN)

GROUP BY companyID, planName

ORDER BY claim\_COUNT s)

SELECT PlanName, CompanyName, Claim\_COUNT s, if(min\_r<=3, 'Least Claimed', 'Most

Claimed') AS Claim\_Category

FROM (SELECT \* FROM Claim\_COUNT WHERE min\_r <=3

UNION

SELECT \* FROM Claim\_COUNT WHERE max\_r <= 3) AS r

JOIN insurancecompany USING(companyID);

**Output 7.3:**

|  |  |  |  |
| --- | --- | --- | --- |
| PlanName | CompanyName | Claim\_Counts | Claim\_Category |
| Surakshit Loan Bima | Future Generali India Insurance Co. Ltd. | 22 | Least Claimed |
| Chola Group Top Up Protect | Cholamandalam MS General | 23 | Least Claimed |
| Star Micro Rural and Farmers Care | Star Health & Allied Insurance Co. Ltd. | 23 | Least Claimed |
| Criti Shield Plus | ICICI Lombard GI Co. Ltd. | 54 | Most Claimed |
| Corona Kavach Policy | Niva Bupa Health Insurance co ltd | 54 | Most Claimed |
| Star Group Critical Illness Multipay Insurance Policy | Star Health and Allied Insurrance Co. | 70 | Most Claimed |
| Group Credit Secure Plus | Tata AIG General Insurance Co. Ltd. | 103 | Most Claimed |

|  |
| --- |
| **Problem Statement 7.4 :**  The healthcare department wants to know which category of patients is being affected the most by each disease. Assist the department in creating a report regarding this. Provided the healthcare department has categorized the patients into the following category.  YoungMale: Born on or after 1st Jan 2005 and gender male.  YoungFemale: Born on or after 1st Jan 2005 and gender female.  AdultMale: Born before 1st Jan 2005 but on or after 1st Jan 1985 and gender male.  AdultFemale: Born before 1st Jan 2005 but on or after 1st Jan 1985 and gender female.  MidAgeMale: Born before 1st Jan 1985 but on or after 1st Jan 1970 and gender male.  MidAgeFemale: Born before 1st Jan 1985 but on or after 1st Jan 1970 and gender female.  ElderMale: Born before 1st Jan 1970, and gender male.  ElderFemale: Born before 1st Jan 1970, and gender female. |

**Solution 7.4 :**

DELIMITER //

CREATE FUNCTION Person\_Category(dob date, gender varchar(50))

RETURNS VARCHAR(20)

DETERMINISTIC

BEGIN

Declare Category varchar(20);

if dob > '2005-01-01' AND gender = 'Male' THEN set category = 'YoungMale';

elseif dob > '2005-01-01' AND gender = 'Female' THEN set category = 'YoungFemale';

elseif dob > '1985-01-01' AND gender = 'Male' THEN set category = 'AdultMale';

elseif dob > '1985-01-01' AND gender = 'Female' THEN set category = 'AdultFemale';

elseif dob > '1970-01-01' AND gender = 'Male' THEN set category = 'MidAgeMale';

elseif dob > '1970-01-01' AND gender = 'Female' THEN set category = 'MidAgeFemale';

elseif dob < '1970-01-01' AND gender = 'Male' THEN set category = 'ElderMale';

elseif dob < '1970-01-01' AND gender = 'Female' THEN set category = 'ElderFemale';

END if;

return (category);

END //

delimiter ;

with patient\_info AS

(SELECT diseaseID, Person\_Category(dob, gender) AS person\_category,

COUNT (patientID) AS patient\_COUNT , rank() over(partition by diseaseID

ORDER BY COUNT (patientID)) AS min\_r, rank() over(partition by diseaseID

ORDER BY COUNT (patientID) desc) AS max\_r

FROM treatment

JOIN patient USING(patientID)

JOIN person on person.personID = patient.patientID

GROUP BY diseaseID,person\_category)

SELECT \*

FROM (SELECT diseaseID, group\_concat(person\_category SEPARATOR ', ') AS

person\_category\_min, patient\_COUNT AS min\_patient\_COUNT

FROM patient\_info

WHERE min\_r = 1

GROUP BY diseaseID, patient\_COUNT ) AS mi

JOIN (SELECT diseaseID, group\_concat(person\_category SEPARATOR ', ') AS

person\_category\_max, patient\_COUNT AS max\_patient\_COUNT

FROM patient\_info

WHERE max\_r = 1

GROUP BY diseaseID, patient\_COUNT ) AS ma USING(diseaseID);

**Output 7.4:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| diseaseID | person\_category\_min | min\_patient\_count | person\_category\_max | max\_patient\_count |
| 1 | AdultFemale | 6 | ElderMale | 85 |
| 2 | AdultFemale | 4 | ElderMale | 84 |
| 3 | AdultFemale | 4 | ElderMale | 104 |
| 4 | AdultMale | 8 | ElderMale | 80 |
| 5 | AdultMale, AdultFemale | 7 | ElderMale | 89 |
| 6 | AdultFemale | 6 | ElderMale | 88 |
| 7 | AdultMale | 6 | ElderMale | 85 |
| 8 | AdultFemale | 1 | ElderMale | 85 |
| 9 | AdultFemale | 14 | ElderMale | 87 |
| 10 | AdultFemale | 7 | ElderMale | 90 |
| 11 | AdultMale | 7 | ElderMale | 107 |
| 12 | AdultFemale | 6 | ElderMale | 85 |
| 13 | AdultFemale | 7 | ElderMale | 82 |
| 14 | AdultFemale | 9 | ElderMale | 96 |

|  |
| --- |
| **Problem Statement 7.5 :**  Anna wants a report on the pricing of the medicine. She wants a list of the most expensive and most affordable medicines only. Assist anna by creating a report of all the medicines which are pricey and affordable, listing the companyName, productName, description, maxPrice, and the price category of each. Sort the list in descending order of the maxPrice. Note: A medicine is considered to be “pricey” if the max price exceeds 1000 and “affordable” if the price is under 5. Write a query to find |

**Solution 7.5 :**

SELECT companyName, productName, description, maxPrice,

if(maxprice>1000, 'Pricey', 'Affordable') AS 'Price\_Category'

FROM medicine

WHERE maxPrice > 1000 or maxPrice < 5

ORDER BY maxPrice desc;

**Output 7.5:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| companyName | productName | description | maxPrice | Price\_Category |
| BIOGEN BRASIL PRODUTOS FARMACEUTICOS LTDA | Spinraza | 2.4 MG/ML SOL INJ CT FA VD TRANS X 5 ML | 388301 | Pricey |
| NATCOFARMA DO BRASIL LTDA | Everolimo | 10 MG COM CT BL AL PLAS PVC TRANS X 1400 | 378664 | Pricey |
| ALEXION FARMACEUTICA BRASIL IMPORTACAO E DISTRIBUICAO DE PRODUTOS E SERVICOS DE ADMINISTRACAO | Strensiq | 100 MG/ML SOL INJ CT 12 FA VD INC X 0.8 ML | 281305.2 | Pricey |
| BLAU FARMACEUTICA S.A. | BOTULIM | 200 U PO LIOF INJ 01 CT C/ 100 FR AMP INC (EMB HOSP) | 240526.5 | Pricey |
| BLAU FARMACEUTICA S.A. | IMUNOGLOBULIN | 50 MG/ML SOL INJ CX 100 FA VD INC X 200 ML | 240467.2 | Pricey |
| NATCOFARMA DO BRASIL LTDA | Everolimo | 5 MG COM CT BL AL PLAS PVC TRANS X 1400 | 189331.7 | Pricey |
| BLANVER FARMOQUIMICA E FARMACEUTICA S.A. | SOFOSBUVIR | 400 MG COM REV CX 50 FR PLAS OPC X 28 | 168077.7 | Pricey |

Solution 8

SQL Query Optimization

|  |
| --- |
| **Problem Statement 8.1 :**  Query 1:  -- For each age(in years), how many patients have gone for treatment?  SELECT DATEDIFF(hour, dob , GETDATE())/8766 AS age, COUNT (\*) AS numTreatments  FROM Person  JOIN Patient ON Patient.patientID = Person.personID  JOIN Treatment ON Treatment.patientID = Patient.patientID  GROUP BY DATEDIFF(hour, dob , GETDATE())/8766  ORDER BY numTreatments desc; |

**Solution 8.1 :**

SELECT concat(round(DATEDIFF( current\_date() , dob)/365.25,-1), ' - ', round(DATEDIFF(

current\_date() , dob)/365.25,-1)+10) AS age\_group, COUNT (\*) AS numTreatments

FROM Patient

JOIN Treatment ON Treatment.patientID = Patient.patientID

GROUP BY age\_group

ORDER BY numTreatments desc;

**Output 8.1:**

|  |  |
| --- | --- |
| age\_group | numTreatments |
| 60 - 70 | 2261 |
| 70 - 80 | 2030 |
| 50 - 60 | 1920 |
| 10-20 | 1514 |
| 40 - 50 | 804 |
| 0 - 10 | 754 |
| 80 - 90 | 664 |
| 20 - 30 | 347 |
| 90 - 100 | 320 |
| 30 - 40 | 230 |
| 100 - 110 | 41 |

|  |
| --- |
| **Problem Statement 8.2 :**  \*Query 2:  -- For each city, Find the number of registered people, number of pharmacies, AND number of insurance companies.  drop table if exists T1;  drop table if exists T2;  drop table if exists T3;  SELECT Address.city, COUNT (Pharmacy.pharmacyID) AS numPharmacy  into T1 FROM Pharmacy right JOIN Address on Pharmacy.addressID = Address.addressID GROUP BY city ORDER BY COUNT (Pharmacy.pharmacyID) desc;  SELECT Address.city, COUNT (InsuranceCompany.companyID) AS numInsuranceCompany into T2 FROM InsuranceCompany right JOIN Address on InsuranceCompany.addressID = Address.addressID GROUP BY city ORDER BY COUNT (InsuranceCompany.companyID) desc;  SELECT Address.city, COUNT (Person.personID) AS numRegisteredPeople  into T3 FROM Person right JOIN Address on Person.addressID = Address.addressID GROUP BY city ORDER BY COUNT (Person.personID) desc;  SELECT T1.city, T3.numRegisteredPeople, T2.numInsuranceCompany, T1.numPharmacy FROM T1, T2, T3 WHERE T1.city = T2.city AND T2.city = T3.city ORDER BY numRegisteredPeople desc; |

**Solution 8.2 :**

-- remove right join,

with pharmacy\_COUNT AS

(SELECT city, COUNT (pharmacyID) AS numPharmacy

FROM Pharmacy JOIN Address USING(addressID)

GROUP BY city

ORDER BY numPharmacy desc),

Insurance\_company\_COUNT AS

(SELECT city, COUNT (companyID) AS numInsuranceCompany

FROM InsuranceCompany JOIN Address USING(addressID)

GROUP BY city

ORDER BY numInsuranceCompany desc),

Registered\_people\_COUNT AS

(SELECT city, COUNT (personID) AS numRegisteredPeople

FROM Person JOIN Address USING(addressID)

GROUP BY city

ORDER BY numRegisteredPeople desc)

SELECT city, coalesce(numRegisteredPeople,0) AS numRegisteredPeople,

coalesce(numInsuranceCompany,0) AS numInsuranceCompany,

coalesce(numInsuranceCompany,0) AS numInsuranceCompany

FROM (SELECT distinct city FROM address) AS city\_name

left JOIN Registered\_people\_COUNT USING(city)

left JOIN pharmacy\_COUNT USING (city)

left JOIN Insurance\_company\_COUNT USING(city)

ORDER BY numRegisteredPeople desc;FROM medicine

WHERE maxPrice > 1000 or maxPrice < 5

ORDER BY maxPrice desc;

**Output 8.2:**

|  |  |  |  |
| --- | --- | --- | --- |
| city | numRegisteredPeople | numInsuranceCompany | numInsuranceCompany |
| Washington | 184 | 4 | 4 |
| Montgomery | 176 | 1 | 1 |
| Manchester | 168 | 2 | 2 |
| Glendale | 153 | 3 | 3 |
| Fayetteville | 149 | 0 | 0 |
| Arvada | 144 | 5 | 5 |
| Anchorage | 135 | 1 | 1 |
| Savannah | 132 | 4 | 4 |
| Louisville | 131 | 3 | 3 |
| Nashville | 130 | 1 | 1 |
| Panama City | 95 | 3 | 3 |
| Oklahoma City | 81 | 1 | 1 |
| Annapolis | 35 | 0 | 0 |
| Panama City Beach | 31 | 0 | 0 |
| Oakland | 31 | 0 | 0 |
| Fremont | 26 | 0 | 0 |
| Glen Burnie | 23 | 1 | 1 |
| Hayward | 21 | 1 | 1 |
| Edmond | 18 | 0 | 0 |

|  |
| --- |
| **Problem Statement 8.3 :**  -- Total quantity of medicine for each prescription prescribed by Ally Scripts  -- If the total quantity of medicine is less than 20 tag it AS "Low Quantity".  -- If the total quantity of medicine is FROM 20 to 49 (both numbers including) tag it AS "Medium Quantity".  -- If the quantity is more than equal to 50 THEN tag it AS "High quantity".  SELECT  C.prescriptionID, sum(quantity) AS totalQuantity,  CASE WHEN sum(quantity) < 20 THEN 'Low Quantity'  WHEN sum(quantity) < 50 THEN 'Medium Quantity'  ELSE 'High Quantity' END AS Tag  FROM Contain C  JOIN Prescription P  on P.prescriptionID = C.prescriptionID  JOIN Pharmacy on Pharmacy.pharmacyID = P.pharmacyID  WHERE Pharmacy.pharmacyName = 'Ally Scripts'  GROUP BY C.prescriptionID; |

**Solution 8.3 :**

SELECT prescriptionID, sum(quantity) AS totalQuantity,

CASE

WHEN sum(quantity) < 20 THEN 'Low Quantity'

WHEN sum(quantity) < 50 THEN 'Medium Quantity'

ELSE 'High Quantity'

END AS Tag

FROM Contain

JOIN Prescription USING(prescriptionID)

WHERE PharmacyID = (SELECT pharmacyID FROM pharmacy

WHERE pharmacyName = 'Ally Scripts')

GROUP BY prescriptionID;

**Output 8.3:**

|  |  |  |  |
| --- | --- | --- | --- |
| prescriptionID | totalQuantity | Tag |  |
| 1147561399 | 43 | Medium Quantity | |
| 1222719376 | 71 | High Quantity | |
| 1408276190 | 48 | Medium Quantity | |
| 1571383871 | 22 | Medium Quantity | |
| 1668180798 | 52 | High Quantity | |
| 1767949601 | 7 | Low Quantity | |
| 1838926888 | 47 | Medium Quantity | |
| 1840130817 | 66 | High Quantity | |
| 1895387326 | 41 | Medium Quantity | |
| 1937085005 | 93 | High Quantity | |

|  |
| --- |
| **Problem Statement 8.4 :**  -- The total quantity of medicine in a prescription is the sum of the quantity of all the medicines in the prescription.  -- SELECT the prescriptions for which the total quantity of medicine exceeds  -- the avg of the total quantity of medicines for all the prescriptions.  drop table if exists T1;  SELECT Pharmacy.pharmacyID, Prescription.prescriptionID, sum(quantity) AS totalQuantity  into T1  FROM Pharmacy  JOIN Prescription on Pharmacy.pharmacyID = Prescription.pharmacyID  JOIN Contain on Contain.prescriptionID = Prescription.prescriptionID  JOIN Medicine on Medicine.medicineID = Contain.medicineID  JOIN Treatment on Treatment.treatmentID = Prescription.treatmentID  WHERE YEAR(date) = 2022  GROUP BY Pharmacy.pharmacyID, Prescription.prescriptionID  ORDER BY Pharmacy.pharmacyID, Prescription.prescriptionID;  SELECT \* FROM T1  WHERE totalQuantity > (SELECT avg(totalQuantity) FROM T1); |

**Solution 8.4 :**

with medicine\_quantity AS

(SELECT pharmacyID, prescriptionID, sum(quantity) AS totalQuantity

FROM Prescription

JOIN Contain USING(prescriptionID)

JOIN treatment USING(treatmentID)

WHERE YEAR(date) = 2022

GROUP BY pharmacyID, prescriptionID

ORDER BY pharmacyID, prescriptionID)

SELECT \* FROM medicine\_quantity

WHERE totalQuantity > (SELECT avg(totalQuantity) FROM medicine\_quantity);

**Output 8.4:**

|  |  |  |  |
| --- | --- | --- | --- |
| prescriptionID | totalQuantity | Tag |  |
| 1147561399 | 43 | Medium Quantity | |
| 1222719376 | 71 | High Quantity | |
| 1408276190 | 48 | Medium Quantity | |
| 1571383871 | 22 | Medium Quantity | |
| 1668180798 | 52 | High Quantity | |
| 1767949601 | 7 | Low Quantity | |
| 1838926888 | 47 | Medium Quantity | |
| 1840130817 | 66 | High Quantity | |
| 1895387326 | 41 | Medium Quantity | |
| 1937085005 | 93 | High Quantity | |

|  |
| --- |
| **Problem Statement 8.5 :**  -- SELECT every disease that hAS 'p' in its name, AND  -- the number of times an insurance claim wAS made for each of them.  SELECT Disease.diseaseName, COUNT (\*) AS numClaims  FROM Disease  JOIN Treatment ON Disease.diseaseID = Treatment.diseaseID  JOIN Claim On Treatment.claimID = Claim.claimID  WHERE diseaseName IN (SELECT diseaseName FROM Disease WHERE diseaseName LIKE '%p%')  GROUP BY diseaseName; |

**Solution 8.5 :**

SELECT diseaseName, COUNT (distinct claimID) AS numClaims

FROM Disease

JOIN Treatment USING(diseaseID)

WHERE diseaseName rlike 'p'

GROUP BY diseaseName;

**Output 8.5:**

|  |  |
| --- | --- |
| diseaseName | numClaims |
| Amyotrophic lateral sclerosis | 167 |
| Attention deficit hyperactivity disorder | 173 |
| Bipolar disorder | 180 |
| Chronic obstructive pulmonary disease | 169 |
| Depression | 158 |
| Diabetes mellitus type 1 | 184 |
| Diabetes mellitus type 2 | 189 |
| Dilated cardiomyopathy | 194 |
| Epilepsy | 141 |
| Low back pain | 169 |
| Lupus | 152 |
| Multiple sclerosis | 165 |
| Obsessive compulsive disorder | 169 |
| Panic disorder | 178 |
| Parkinson's disease | 150 |
| Psoriasis | 167 |
| Schizophrenia | 200 |

Solution 9

SQL Rollup

|  |
| --- |
| **Problem Statement 9.1 :**  Brian, the healthcare department, has requested for a report that shows for each state how many people underwent treatment for the disease “Autism”.  He expects the report to show the data for each state as well as each gender and for each state and gender combination. Prepare a report for Brian for his requirement. |

**Solution 9.1 :**

SELECT coalesce(State, 'Total') AS 'State', coalesce(gender, 'Total') AS 'Gender',

COUNT(patientID) AS 'Patient\_COUNT '

FROM disease

JOIN treatment USING(diseaseID)

JOIN person on person.personID = treatment.patientID

JOIN address USING(addressID)

WHERE diseaseName = 'Autism'

GROUP BY state, gender with rollup;

**Output 9.1:**

|  |  |  |
| --- | --- | --- |
| State | Gender | Patient\_Count |
| AK | female | 7 |
| AK | male | 9 |
| AK | Total | 16 |
| AL | female | 6 |
| AL | male | 12 |
| AL | Total | 18 |
| AR | female | 6 |
| AR | male | 10 |
| AR | Total | 16 |
| AZ | female | 6 |
| AZ | male | 7 |
| AZ | Total | 13 |
| CA | female | 5 |
| CA | male | 15 |
| CA | Total | 20 |
| CO | female | 7 |
| CO | male | 11 |
| CO | Total | 18 |
| CT | female | 6 |

|  |
| --- |
| **Problem Statement 9.2 :**  Insurance companies want to evaluate the performance of different insurance plans they offer. Generate a report that shows each insurance plan, the company that issues the plan, AND the number of treatments the plan was claimed for. The report would be more relevant if the data compares the performance for different years(2020, 2021 AND 2022) AND if the report also includes the total number of claims in the different years, AS well AS the total number of claims for each plan in all 3 years combined. |

**Solution 9.2 :**

with claim\_info as

(select planName, companyName, coalesce(year(date),'total') as 'Year',

count(claimID) as Claim\_Count

from insurancecompany

join insuranceplan using(companyID)

join claim using(UIN)

join treatment using(claimID)

where year(date) in (2020,2021,2022)

group by PlanName, companyName, year(date) with rollup)

select \* from claim\_info where (companyName is not null) and (planName is not null);

**Output 9.2:**

|  |  |  |  |
| --- | --- | --- | --- |
| planName | companyName | Year | Claim\_Count |
| Additional S.I. for Pandemic and epidemic Rider | Aditya Birla Health Insurance Co. Ltd | 2020 | 7 |
| Additional S.I. for Pandemic and epidemic Rider | Aditya Birla Health Insurance Co. Ltd | 2021 | 6 |
| Additional S.I. for Pandemic and epidemic Rider | Aditya Birla Health Insurance Co. Ltd | 2022 | 10 |
| Additional S.I. for Pandemic and epidemic Rider | Aditya Birla Health Insurance Co. Ltd | total | 23 |
| Alpa Bima ï¿½ Group | Future Generali India Insurance Co. Ltd. | 2020 | 12 |
| Alpa Bima ï¿½ Group | Future Generali India Insurance Co. Ltd. | 2021 | 8 |
| Alpa Bima ï¿½ Group | Future Generali India Insurance Co. Ltd. | 2022 | 8 |
| Alpa Bima ï¿½ Group | Future Generali India Insurance Co. Ltd. | total | 28 |
| Arogya Premier Policy | SBI General Insurance Co.Ltd. | 2020 | 12 |
| Arogya Premier Policy | SBI General Insurance Co.Ltd. | 2021 | 6 |
| Arogya Premier Policy | SBI General Insurance Co.Ltd. | 2022 | 7 |
| Arogya Premier Policy | SBI General Insurance Co.Ltd. | total | 25 |

|  |
| --- |
| **Problem Statement 9.3 :**  Sarah, from the healthcare department, is trying to understand if some diseases are spreading in a particular region. Assist Sarah by creating a report which shows each state the number of the most and least treated diseases by the patients of that state in the year 2022. It would be helpful for Sarah if the aggregation for the different combinations is found as well. Assist Sarah to create this report. |

**Solution 9.3 :**

with cte as

(select state, diseaseID, count(patientID) as 'treatment\_count',

rank() over(partition by state order by count(patientID)) as min\_r,

rank() over(partition by state order by count(patientID) desc) as max\_r

from treatment

join disease using(diseaseID)

join person on treatment.patientID = person.personID

join address using(addressID)

group by state, diseaseID)

select \* from cte

where min\_r = 1 or max\_r = 1;

**Output 9.3:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| state | diseaseID | treatment\_count | min\_r | max\_r |
| AK | 21 | 17 | 40 | 1 |
| AK | 15 | 5 | 1 | 40 |
| AL | 11 | 28 | 39 | 1 |
| AL | 22 | 28 | 39 | 1 |
| AL | 13 | 9 | 1 | 40 |
| AR | 35 | 22 | 38 | 1 |
| AR | 4 | 22 | 38 | 1 |
| AR | 36 | 22 | 38 | 1 |
| AR | 21 | 9 | 1 | 38 |
| AR | 38 | 9 | 1 | 38 |
| AR | 17 | 9 | 1 | 38 |
| AZ | 20 | 24 | 40 | 1 |
| AZ | 33 | 9 | 1 | 40 |
| CA | 6 | 40 | 40 | 1 |
| CA | 29 | 18 | 1 | 40 |
| CO | 17 | 26 | 40 | 1 |
| CO | 13 | 9 | 1 | 40 |
| CT | 28 | 27 | 40 | 1 |
| CT | 32 | 5 | 1 | 40 |

|  |
| --- |
| **Problem Statement 9.4 :**  Jackson has requested a detailed pharmacy report that shows each pharmacy name, and how many prescriptions they have prescribed for each disease in the year 2022, along with this Jackson also needs to view how many prescriptions were prescribed by each pharmacy, and the total number prescriptions were prescribed for each disease. Assist Jackson to create this report. |

**Solution 9.4 :**

SELECT pharmacyName, coalesce(DiseaseName, 'Total') AS Disease\_Name,

COUNT (prescriptionID) AS 'Prescription\_COUNT '

FROM Pharmacy

JOIN Prescription USING(pharmacyID)

JOIN treatment USING(treatmentID)

JOIN disease USING(diseaseID)

WHERE year(date) = 2022

GROUP BY pharmacyName, DiseaseName with rollup

ORDER BY pharmacyName, Prescription\_COUNT desc;

**Output 9.4:**

|  |  |  |
| --- | --- | --- |
| pharmacyName | DiseaseName | Prescription\_Count |
| Absolute Care | Total | 13 |
| Absolute Care | Epilepsy | 2 |
| Absolute Care | Autoimmune diseases | 2 |
| Absolute Care | Asthma | 1 |
| Absolute Care | Depression | 1 |
| Absolute Care | Diabetes mellitus type 1 | 1 |
| Absolute Care | Diabetes mellitus type 2 | 1 |
| Absolute Care | Dilated cardiomyopathy | 1 |
| Absolute Care | Anorexia nervosa | 1 |
| Absolute Care | Guillainarr syndrome | 1 |
| Absolute Care | Irritable bowel syndrome | 1 |
| Absolute Care | Thromboangiitis obliterans | 1 |

|  |
| --- |
| **Problem Statement 9.5 :**  Praveen has requested for a report that finds for every disease how many males and females underwent treatment for each in the year 2022. It would be helpful for Praveen if the aggregation for the different combinations is found as well. Assist Praveen to create this report. |

**Solution 9.5 :**

SELECT diseaseName, coalesce(gender, 'Total') AS 'Gender',

COUNT(patientID) AS 'Patient\_COUNT '

FROM disease

JOIN treatment USING(diseaseID)

JOIN person on person.personiD= treatment.patientID

GROUP BY diseaseName, gender with rollup;

**Output 9.5:**

|  |  |  |
| --- | --- | --- |
| diseaseName | Gender | Patient\_Count |
| Alzheimer's disease | female | 95 |
| Alzheimer's disease | male | 173 |
| Alzheimer's disease | Total | 268 |
| Amyotrophic lateral sclerosis | female | 106 |
| Amyotrophic lateral sclerosis | male | 165 |
| Amyotrophic lateral sclerosis | Total | 271 |
| Anorexia nervosa | female | 96 |
| Anorexia nervosa | male | 177 |
| Anorexia nervosa | Total | 273 |
| Anxiety disorder | female | 126 |
| Anxiety disorder | male | 153 |
| Anxiety disorder | Total | 279 |
| Asthma | female | 101 |
| Asthma | male | 144 |
| Asthma | Total | 245 |
| Atherosclerosis | female | 112 |
| Atherosclerosis | male | 174 |
| Atherosclerosis | Total | 286 |
| Attention deficit hyperactivity disorder | female | 125 |
| Attention deficit hyperactivity disorder | male | 158 |
| Attention deficit hyperactivity disorder | Total | 283 |

Solution 10

SQL Stored Routines

|  |
| --- |
| **Problem Statement 10.1 :**  The healthcare department has requested a system to analyze the performance of insurance companies and their plan.For this purpose, create a stored procedure that returns the performance of different insurance plans of an insurance company. When passed the insurance company ID the procedure should generate and return all the insurance plan names the provided company issues, the number of treatments the plan was claimed for, and the name of the disease the plan was claimed for the most. The plans which are claimed more are expected to appear above the plans that are claimed less. |

**Solution 10.1 :**

DELIMITER //

Create Procedure Company\_Plans\_info(company\_id\_in int)

Begin

with plans\_info AS

(SELECT planName, claimID, diseaseID

FROM insurancePlan

JOIN claim USING(UIN)

JOIN treatment USING(claimID)

WHERE companyID = company\_id\_in),

claim\_COUNT \_info as

(SELECT planName, COUNT (claimID) AS 'Claim\_COUNT '

FROM plans\_info GROUP BY planName),

disease\_id\_info AS

(SELECT planName, diseaseID, COUNT (claimID),

rank() over(Partition by planName ORDER BY COUNT (claimID) desc)

AS 'max\_r'

FROM plans\_info GROUP BY planName, diseaseID

ORDER BY planName, COUNT (claimID) desc),

disease\_Name\_Info as

(SELECT planName, group\_concat(diseaseName separator ', ') AS

'Max\_Claimed\_Disease'

FROM disease\_ID\_info

JOIN disease USING(diseaseID)

WHERE max\_r = 1 GROUP BY planName)

SELECT \* FROM disease\_Name\_info JOIN claim\_COUNT \_info USING(planName);

END //

Delimiter ;

call Company\_Plans\_info(6403);

**Output 10.1:**

|  |  |  |
| --- | --- | --- |
| planName | Max\_Claimed\_Disease | Claim\_count |
| Bajaj Allianz Bharat Bhraman Insurance Policy | Obesity | 37 |
| E Travel Value policy | Diabetes mellitus type 2 | 36 |
| Group Business Travelers Insurance Policy | Epilepsy, Stroke, Coronary heart disease | 31 |
| Group Hospital Cash Policy | Tourette syndrome | 33 |
| IRCTC Air Care Policy | Tourette syndrome | 37 |
| Pravasi Bharatiya Bima Yojana Polic | Diabetes mellitus type 1, Anxiety disorder | 47 |
| Travel Assist Cardï¿½ | Dilated cardiomyopathy | 38 |
| Travel Companion | Alzheimer's disease, Low back pain | 29 |
| Travel Elect (Group) | Chronic obstructive pulmonary disease | 41 |
| Travel Insurance for E- ticket Passengers-For(IRCTC) | Lupus, Chronic fatigue syndrome, Guillain syndrome, Depression | 42 |
| Travel Prime Holiday Insurance Policy (Group) | Guillain syndrome | 27 |
| Travel Prime Policy | Panic disorder | 36 |

|  |
| --- |
| **Problem Statement 10.2 :**  It was reported by some unverified sources that some pharmacies are more popular for certain diseases. The healthcare department wants to check the validity of this report. Create a stored procedure that takes a disease name as a parameter and would return the top 3 pharmacies the patients are preferring for the treatment of that disease in 2021 AS well as for 2022. Check if there are common pharmacies in the top 3 list for a disease, in the years 2021 and the year 2022. Call the stored procedure by passing the values “Asthma” and “Psoriasis” as disease names and draw a conclusion from the result. |

**Solution 10.2 :**

Delimiter //

Create procedure Disease\_Pharmacy\_info(in Pharmacy\_name\_in varchar(50))

Begin

SELECT \*

FROM (SELECT pharmacyID, pharmacyName,

COUNT (treatmentID) AS 'Treatment\_COUNT ',

rank() over(ORDER BY COUNT (patientID) desc) AS 'max\_rank'

FROM disease

JOIN treatment USING(diseaseID)

JOIN prescription USING(treatmentID)

JOIN pharmacy USING(pharmacyID)

WHERE diseaseName = Pharmacy\_name\_in

GROUP BY pharmacyID) AS q1

WHERE max\_rank <=3;

END //

Delimiter ;

call Disease\_Pharmacy\_info('Asthma') ;

**Output 10.2.1:**

|  |  |  |  |
| --- | --- | --- | --- |
| pharmacyID | pharmacyName | Treatment\_count | max\_rank |
| 7016 | Southside Family Pharmacy | 5 | 1 |
| 9659 | Express Scripts | 5 | 1 |
| 8760 | Discount Drugs | 4 | 3 |
| 3673 | Pharmacy Express | 4 | 3 |
| 6330 | North East Pharmacy | 4 | 3 |
| 9707 | Better You | 4 | 3 |
| 9296 | Gem Drugs | 4 | 3 |
| 2245 | Rxtra | 4 | 3 |
| 2593 | Health Harvest | 4 | 3 |
| 8824 | Holloway Pharmacy | 4 | 3 |
| 8737 | RX Express | 4 | 3 |
| 3563 | Vitasource | 4 | 3 |
| 1925 | Everyday Drugs | 4 | 3 |
| 9999 | Planet Health | 4 | 3 |
| 1574 | ScriptSite Specialty | 4 | 3 |
| 4647 | Goodstart | 4 | 3 |

call Disease\_Pharmacy\_info('Psoriasis');

**Output 10.2.2:**

|  |  |  |  |
| --- | --- | --- | --- |
| pharmacyID | pharmacyName | Treatment\_count | max\_rank |
| 1354 | HealthMart | 5 | 1 |
| 1584 | Sunwest | 5 | 1 |
| 2836 | Discount Drug Mart | 4 | 3 |
| 9255 | PrecisionMed | 4 | 3 |
| 2563 | PharmaMed | 4 | 3 |
| 3628 | Midtown Express | 4 | 3 |
| 4749 | Lifechek | 4 | 3 |
| 4326 | Pharmanic | 4 | 3 |
| 4663 | Tru Script | 4 | 3 |
| 4106 | Central Rx | 4 | 3 |
| 4891 | Medisuite | 4 | 3 |
| 3673 | Pharmacy Express | 4 | 3 |

|  |
| --- |
| **Problem Statement 10.3 :**  Jacob, as a business strategist, wants to figure out if a state is appropriate for setting up an insurance company or not. Write a stored procedure that finds the num\_patients, num\_insurance\_companies, and insurance\_patient\_ratio, the stored procedure should also find the avg\_insurance\_patient\_ratio and if the insurance\_patient\_ratio of the given state is less than the avg\_insurance\_patient\_ratio then it Recommendation section can have the value “Recommended” otherwise the value can be “Not Recommended”. Description of the terms used:  num\_patients: number of registered patients in the given state  num\_insurance\_companies: The number of registered insurance companies in the given state  insurance\_patient\_ratio: The ratio of registered patients and the number of insurance companies in the given state  avg\_insurance\_patient\_ratio: The average of the ratio of registered patients and the number of insurance for all the states. |

**Solution 10.3 :**

Delimiter //

create procedure State\_Recommendation(in state\_name varchar(10))

begin

set @avg\_insurance\_patient\_ratio = (SELECT COUNT (distinct patientID)

FROM patient)/(SELECT COUNT (distinct companyID) FROM

insurancecompany);

with patient\_COUNT AS

(SELECT state, COUNT (distinct patientID) AS num\_patients

FROM (SELECT \* FROM address WHERE state = state\_name) AS address

JOIN person USING(addressID)

JOIN patient on person.personID = patient.patientID),

company\_COUNT AS

(SELECT state, COUNT (distinct companyID) AS num\_insurance\_companies

FROM (SELECT \* FROM address WHERE state = state\_name) AS address

JOIN insurancecompany USING(addressID))

SELECT \*, (num\_patients/num\_insurance\_companies) AS 'insurance\_patient\_ratio',

@avg\_insurance\_patient\_ratio AS avg\_insurance\_patient\_ratio,

if (num\_patients/num\_insurance\_companies < @avg\_insurance\_patient\_ratio, 'Recommended', 'Not Recommended') AS 'Recommendation'

FROM patient\_COUNT JOIN company\_COUNT USING(state);

END //

delimiter ;

call State\_Recommendation('OK');

**Output 10.3:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| state | num\_patients | num\_insurance\_companies | insurance\_patient\_ratio | avg\_insurance\_patient\_ratio | Recommendation |
| OK | 65 | 2 | 32.5 | 26.18605 | Not Recommended |

|  |
| --- |
| **Problem Statement 10.4 :**  Currently, the data FROM every state is not in the database, The management has decided to ADD the data FROM other states AND cities AS well. It is felt by the management that it would be helpful if the date AND time were to be stored whenever new city or state data is inserted. The management has sent a requirement to create a PlacesAdded table if it doesn’t already exist, that has four attributes. placeID, placeName, placeType, and timeAdded.  Description  placeID: This is the primary key, it should be auto-incremented starting FROM 1  placeName: This is the name of the place which is added for the first time  placeType: This is the type of place that is added for the first time. The value can either be ‘city’ or ‘state’  timeAdded: This is the date and time when the new place is added  You have been given the responsibility to create a system that satisfies the requirements of the management. Whenever some data is inserted in the Address table that has a new city or state name, the PlacesAdded table should be updated with relevant data. |

**Solution 10.4 :**

create table if not exists placeadded

(PlaceID int primary key auto\_increment,

PlaceName varchar(20) Not Null,

PlaceType enum('City', 'State'),

TimeAdded datetime);

Delimiter //

CREATE TRIGGER New\_city\_State

Before Insert on Address for each row

Begin

if new.addressID not in (SELECT addressID FROM address) then

if new.state not in (SELECT distinct state FROM address)

THEN insert into placeadded(PlaceName, PlaceType, TimeAdded)

values ( new.state, 'State', now());

END if;

if new.city not in (SELECT distinct city FROM address)

THEN insert into placeadded(PlaceName, PlaceType, TimeAdded)

values ( new.city, 'City', now());

END if;

END if;

END //

delimiter;

INSERT INTO address VALUES

(10933, 'Futurense Technologies', 'Bengaluru', 'KA', 10092),

(32419, 'Moti Dungri', 'Alwar', 'RJ', 301001);

SELECT \* FROM placeadded;

**Output 10.4:**

|  |  |  |  |
| --- | --- | --- | --- |
| PlaceID | PlaceName | PlaceType | TimeAdded |
| 1 | KA | State | 18-05-2023 17:25 |
| 2 | Bengaluru | City | 18-05-2023 17:25 |
| 3 | RJ | State | 18-05-2023 17:25 |
| 4 | Alwar | City | 18-05-2023 17:25 |

|  |
| --- |
| **Problem Statement 10.5 :**  Some pharmacies suspect there is some discrepancy in their inventory management. The quantity in the ‘Keep’ is updated regularly AND there is no record of it. They have requested to create a system that keeps track of all the transactions whenever the quantity of the inventory is updated.You have been given the responsibility to create a system that automatically updates a Keep\_Log table which has the following fields:  id: It is a unique field that starts with 1 AND increments by 1 for each new entry  medicineID: It is the medicineID of the medicine for which the quantity is updated.  quantity: The quantity of medicine which is to be added. If the quantity is reduced THEN the number can be negative.  For example: If in Keep the old quantity was 700 AND the new quantity to be updated is 1000, THEN in Keep\_Log the quantity should be 300.  Example 2: If in Keep the old quantity was 700 AND the new quantity to be updated is 100, THEN in Keep\_Log the quantity should be -600. |

**Solution 10.5:**

create table if not exists Keep\_log

(id int primary key auto\_increment,

medicineID int,

quantity int);

delimiter //

create trigger Medicine\_Update\_log

After UPDATE on keep for each row

Begin

if new.quantity != old.quantity

THEN insert into keep\_log values(default, old.medicineID, new.quantity –

old.quantity);

END if;

END //

delimiter;

UPDATE keep set quantity = 5949 WHERE pharmacyID = 1008 AND medicineID = 1111;

SELECT \* FROM keep\_log;

UPDATE keep SET quantity = 3487 WHERE pharmacyID = 1008 AND medicineID = 1111;

UPDATE keep SET quantity = 9876 WHERE pharmacyID = 1008 AND medicineID = 9202;

SELECT \* FROM keep\_log;

**Output 10.5:**

|  |  |  |
| --- | --- | --- |
| id | medicineID | quantity |
| 1 | 1111 | -2462 |
| 2 | 9202 | 438 |

Solution 11

SQL Stored Routines 2

|  |
| --- |
| **Problem Statement 11.1 :**  Patients are complaining that it is often difficult to find some medicines. They move from pharmacy to pharmacy to get the required medicine. A system is required that finds the pharmacies and their contact number that have the required medicine in their inventory. So that the patients can contact the pharmacy and order the required medicine. Create a stored procedure that can fix the issue. |

**Solution 11.1 :**

Delimiter //

Create Procedure Pharmacy\_Info(Medicine\_Name varchar(50))

begin

SELECT pharmacyName, phone, address1, city, state

FROM pharmacy

JOIN address USING(addressID)

WHERE pharmacyID in (SELECT distinct pharmacyID FROM keep

WHERE medicineID in (SELECT medicineID FROM medicine

WHERE productName = Medicine\_Name));

END //

delimiter ;

SELECT \* FROM medicine;

call pharmacy\_info('CETIL');

**Output 11.1:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| pharmacyName | phone | address1 | city | state |
| PrecisionMed | 6874892953 | 6276 Devinney Circle | Arvada | CO |
| Centrico | 2784601947 | 13583 West 68th Avenue | Arvada | CO |
| Goodstart | 9517743319 | 1324 Beddington Park | Nashville | TN |

|  |
| --- |
| **Problem Statement 11.2 :**  The pharmacies are trying to estimate the average cost of all the prescribed medicines per prescription, for all the prescriptions they have prescribed in a particular year. Create a stored function that will return the required value when the pharmacyID and year are passed to it. Test the function with multiple values. |

**Solution 11.2 :**

Delimiter //

Create function Avg\_Pres\_Price\_Year(pharmacy\_ID\_IN int, year\_in int)

returns int deterministic

Begin

declare Avg\_Prescription\_Price int ;

SELECT round(sum(quantity\*maxPrice)/ COUNT (distinct prescriptionID), 2) into

Avg\_Prescription\_Price

FROM treatment

JOIN prescription USING(treatmentID)

JOIN contain USING(prescriptionID)

JOIN medicine USING(medicineID)

WHERE pharmacyID = pharmacy\_ID\_IN AND

year(date) = year\_in;

return Avg\_Prescription\_Price;

END //

delimiter ;

SELECT distinct pharmacyID, year(date) AS 'Year', Avg\_Pres\_Price\_Year(pharmacyID,

year(date)) AS Avg\_Pres\_Price\_Year

FROM treatment JOIN prescription USING(treatmentID)

WHERE pharmacyID = 1008;

**Output 11.2:**

|  |  |  |
| --- | --- | --- |
| pharmacyID | Year | Avg\_Pres\_Price\_Year |
| 1008 | 2019 | 3416 |
| 1008 | 2020 | 11897 |
| 1008 | 2022 | 30593 |
| 1008 | 2021 | 5830 |

|  |
| --- |
| **Problem Statement 11.3 :**  The healthcare department has requested an application that finds out the disease that was spread the most in a state for a given year. So that they can use the information to compare the historical data and gain some insight. Create a stored function that returns the name of the disease for which the patients from a particular state had the most number of treatments for a particular year. Provided the name of the state and year is passed to the stored function. |

**Solution 11.3 :**

Delimiter //

Create function Most\_Spread\_Disease(State\_Name varchar(10), year\_In int)

returns varchar(150) deterministic

begin

declare res varchar(150);

SELECT group\_concat(diseaseName separator ', ') into res

FROM (SELECT diseaseID, DiseaseName, COUNT (patientID) AS 'Infected\_Patients',

rank() over(ORDER BY COUNT (diseaseID) desc) AS max\_r

FROM disease

JOIN treatment USING(diseaseID)

JOIN person on person.personID = treatment.patientID

JOIN address USING(addressID)

WHERE state = State\_Name AND year(date) = year\_In

GROUP BY diseaseID) AS q1

WHERE max\_r = 1;

return res;

END //

Delimiter ;

SELECT distinct state, year(date), Most\_Spread\_Disease(state, year(date)) AS

Most\_Spread\_Disease

FROM treatment

JOIN person on person.personID = treatment.patientID

JOIN address USING(addressID)

WHERE state = 'VT' ;

**Output 11.3:**

|  |  |  |
| --- | --- | --- |
| state | year(date) | Most\_Spread\_Disease |
| VT | 2020 | Sarcoidosis |
| VT | 2022 | Metabolic syndrome |
| VT | 2021 | Vasculitis, Alzheimer's disease |
| VT | 2019 | Dilated cardiomyopathy |
| VT | 2018 | Myocardial infarction |

|  |
| --- |
| **Problem Statement 11.4 :**  The representative of the pharma union, Aubrey, has requested a system that she can use to find how many people in a specific city have been treated for a specific disease in a specific year. Create a stored function for this purpose. |

**Solution 11.4 :**

Delimiter //

Create function Disease\_in\_City\_in\_year(Disease\_ID\_In int, city\_name varchar(50), year\_In

int)

returns int deterministic

begin

declare patient\_COUNT int;

SELECT COUNT (patientID) into Patient\_COUNT

FROM disease

JOIN treatment USING(diseaseID)

JOIN person on person.personID = treatment.patientID

JOIN address USING(addressID)

WHERE city = city\_name AND diseaseID= disease\_ID\_In AND year(date) = year\_In

GROUP BY city, diseaseID, year(date);

return Patient\_COUNT ;

END //

delimiter ;

SELECT Disease\_in\_City\_in\_year(4,'Glen Burnie',2022) as patient\_count;

**Output 11.4:**

|  |
| --- |
| patient\_count |
| 3 |

|  |
| --- |
| **Problem Statement 11.5 :**  The representative of the pharma union, Aubrey, is trying to audit different aspects of the pharmacies. She has requested a system that can be used to find the average balance for claims submitted by a specific insurance company in the year 2022. Create a stored function that can be used in the requested application. |

**Solution 11.5 :**

delimiter //

create function avg\_bal\_2k22(company\_ID\_In int)

returns int deterministic

begin

declare avg\_balance int ;

SELECT round(avg(balance), 0) into avg\_balance

FROM insuranceplan

JOIN claim USING(UIN)

JOIN treatment USING(claimID)

WHERE companyID = company\_ID\_In AND year(date) = 2022;

return avg\_balance;

END //

delimiter ;

SELECT companyID, avg\_bal\_2k22(companyID) as avg\_bal FROM insurancecompany;

**Output 11.5:**

|  |  |
| --- | --- |
| companyID | avg\_bal |
| 1118 | 555093 |
| 1409 | 610306 |
| 1583 | 520585 |
| 1839 | 496111 |
| 1933 | 521950 |
| 1999 | 416748 |
| 2295 | 557584 |
| 2725 | 474181 |
| 3035 | 569748 |
| 3489 | 460684 |
| 3827 | 440296 |
| 4559 | 415744 |
| 4872 | 548097 |
| 4987 | 422236 |