**CS3431-A23 Wong**

**Assignment 3: SQL 5, Relational Algebra**

σ, π, τ, γ, **δ,** ⋈, 🡨, ∩, **∪**

RT 🡨 ReservedTour (reservedTourID PK, travelDate, customerID FK, tourID FK, guideID FK, price)

C 🡨 Customer (customerID PK, firstName, lastName, address, phone UQ, age)

T 🡨 Tour (tourID PK, tourName, description, city, state, vehicleType, price)

L 🡨 Location (locationID PK, locationName, locationType, address, tourID FK)

G 🡨 Guide (guideID PK, firstName, lastName, driverLicense, title, salary, licenseType)

**For the questions below, write the relational algebra. Use the abbreviations for the tables or 5 points off.**

1. (10 points) List tours that have customers who are over 65 OR tours that use boats. Include the travel date, customer first name and last name as a single field called ‘fullName’, customer age, and tour name. Sort by tour name and then by customer full names. Use theta joins. This is from Assignment 1 Part 2.d.
   1. τ tourname, fullname πtravel date, firstname ||’ ‘|| lastname as Fullname, age, tourname σ t.vehicleType=boat(T) ⋈t.tourid=rt.tourid RT ⋈rt.customerid=c.customerid σc.age>65(C)
2. (10 points) List for each level of guide - junior guide, guide or senior guide - how many mismatches there are between the required tour’s vehicle type and the guide's license type. Use theta joins. This is from Assignment 2 Part 2 Q1.a.
   1. γtitle, count(licenseType) as Mismatches σ (vehicleType = 'boat' and licenseType = 'land') or (vehicleType = 'car' and licenseType = 'sea') or (vehicleType = 'bus' and licenseType = 'sea') ((T) ⋈t.tourid=rt.tourid RT ⋈rt.guideid=g.guideid (G))
3. (10 points) Our tour company has bought a competitor and we now have additional guides. Their information is in a separate table with the same schema as Guide:

**NG 🡨 NewGuide** (newGuideID PK, firstName, lastName, driverLicense, title, salary, licenseType)

For all guides (including the new ones), display for each guide title the sum of the salaries being paid.

* 1. γtitle, sum(salary) (G **∪** NG)

1. (10 points) Determine the customer who will make the most number of visits to tour locations. List the firstName, lastName, and the number of location visits (use the heading Visits). It may help to look at the PowerPoint slides at the end of SQL 5.
   1. **T1 🡨**γc.customerid πcount(l.locationid) as numVisits **C**⋈c.customerid=rt.customerid **RT** ⋈rt.tourid=l.tourid **L**
   2. **T2 🡨** πmax(numvisit)**T1**
   3. γc.firstName, c.lastName π c.firstName, c.lastName, count(l.locationid) as Visits σ count(l.locationid)=**T2****C**⋈c.customerid=rt.customerid **RT** ⋈rt.tourid=l.tourid **L**
2. (10 points) Given the schema below:

**SF 🡨 SFBook** (ISBN PK, title, year, price, awardWinner, publisherName) -- science fiction books

**S 🡨 Stock** (warehouseCode PK, ISBN PK FK, city, numberOfBooks)

Report the warehouse code and city for warehouses that stock fewer than 10 copies of any book published by the publisher ‘Wiley’. Write efficient relational algebra.

π warehouseCode, city (σ publisherName=’Wiley’ SF)⋈SF.ISBN=S.ISBN (σ numberOfBooks<=10 S)

**Part 2 (10 points): Interpreting Relational Algebra Expressions**

Be careful in doing these two problems as NO partial credit will be given!

1. (5 points) N ⋈A=T ( π A as T (M) - πB as T (N))

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **M** | | | |  | **N** | | |
| **A** | **B** | **Y** | **Z** |  | **A** | **B** | **Q** |
| 3 | 4 | Tom | Jane |  | 6 | 4 | David |
| 6 | 2 | Mary | Susan |  | 7 | 10 | Jane |
| 10 | 8 | David | Paul |  | 4 | 7 | Paul |
| 4 | 3 | Mark | Helen |  | 3 | 1 | Tom |
| 1 | 1 | Lisa | Brian |  | 3 | 2 | Susan |

|  |  |  |  |
| --- | --- | --- | --- |
| N ⋈A=T ( π A as T (M) - πB as T (N)) | | | |
| A | B | Q | T |
| 6 | 4 | David | 6 |
| 4 | 7 | Paul | 4 |

1. (5 points) (γZ, count(B) As G , min(A) As H (M)) ⋈G=A or H=Q N

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **M** | | | |  | **N** | | |
| **A** | **B** | **Y** | **Z** |  | **A** | **B** | **Q** |
| 1 | 4 | α | α |  | 2 | 3 | 4 |
| 3 | 5 | β | β |  | 8 | 2 | 3 |
| 7 | 6 | α | β |  | 2 | 5 | 10 |
| 5 | 8 | β | α |  |  |  |  |
| 13 | 10 | β | β |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| (γZ, count(B) As G , min(A) As H (M)) ⋈G=A or H=Q N | | | | | |
| Z | G | H | A | B | Q |
| α | 2 | 1 | 2 | 3 | 4 |
| β | 3 | 3 | 8 | 2 | 3 |

**Part 3 (40 points): SQL Queries**

Use a text editor to create **tour3.sql** to hold the following SQL solutions. You will want to use the relational algebra you created above to build the SQL queries part-by-part and seeing the intermediate results before putting together the entire query. Create the following SQL queries according to the schemas given in part 1. Assign and use table aliases wherever it is feasible, and particularly when doing theta joins.

1. (15 points) Our tour company has bought a competitor and we now have additional guides. Their information is in a separate table with the same schema as Guide:  
   **NG 🡨 NewGuide** (newGuideID PK, firstName, lastName, driverLicense, title, salary, licenseType)

For all guides (including the new ones), display for each guide title the sum of the salaries being paid. This is the SQL solution to the relational algebra problem in Part 1.3. Note that NewGuide and Guide may have some of the same primary key values.

1. (10 points) Determine the customer who will make the most number of visits to tour locations. List the firstName, lastName, and the number of location visits (use the heading Visits). This is the SQL solution to the relational algebra problem in Part 1.4. It may help to look at the PowerPoint slides at the end of SQL 5.
2. (15 points) Given the schema below:

**SF 🡨 SFBook** (ISBN PK, title, year, price, awardWinner, publisherName) -- science fiction books

**S 🡨 Stock** (warehouseCode PK, ISBN PK FK, city, numberOfBooks)

Report the warehouse code and city for warehouses that stock fewer than 10 copies of any book published by the publisher ‘Wiley’. Write an efficient SQL query. This is the SQL solution to the relational algebra problem in Part 1.5.