**Relational Algebra** (Last updated 2020-02-14)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Relational Algebra**  This is a data manipulation language where all query operations result in relations.  Operations:  Projection chooses a subset of columns from a relation  Selection chooses a subset of rows based on a condition  Join joins rows of two relations  Division checks for all occurrences of the divisor in the dividend  Difference subtracts occurrences of the second relation from the first relation  **Precedence is left-to-right assoicativity, and can be changed with parentheses.** | Consider the following relations:   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | Student   |  |  | | --- | --- | | **ST#** | **Classif** | | 100 | FR | | 200 | SO | | 300 | JR | | 400 | SR | | 500 | FR |   Course   |  |  | | --- | --- | | C# | Title | | CS 1713 | Intro to CS | | MAT 1224 | Calculus I | | ENG 1043 | English I | | HIS 1033 | History of the World Part I | | Enrollment   |  |  |  |  | | --- | --- | --- | --- | | **ST#** | **C#** | **Midterm** | **Final** | | 100 | CS 1713 | 95 | 80 | | 100 | MAT 1224 | 100 | 100 | | 100 | ENG 1043 | 85 | 85 | | 100 | HIS 1033 | 80 | 85 | | 200 | CS 1713 | 91 | 90 | | 200 | MAT 1224 | 50 | 80 | | 300 | CS 1713 | 65 | 80 | | 500 | CS 1713 | 100 | 80 | | 500 | HIS 1033 | 95 | 90 | | |
| **Projection**  Chooses specified columns from a relation.  Syntax: *relation* [ *columnList* ]  For the specified relation, this projects (i.e., chooses) the specified columns from it. The result will have those columns from each row. Since the result is a relation, duplicates are removed. | **Example 1:** **Projection** - Get the C# for courses in which students are enrolled.  Enrollment[C#]   |  | | --- | | **C#** | | CS 1713 | | MAT 1224 | | ENG 1043 | | HIS 1033 |   **Example 2:** **Projection** - Get the ST# and Midterm grades of each student.  Enrollment[ST#, Midterm]   |  |  | | --- | --- | | **ST#** | **Midterm** | | 100 | 95 | | 100 | 100 | | 100 | 85 | | 100 | 80 | | 200 | 91 | | 200 | 50 | | 300 | 65 | | 500 | 100 | | 500 | 95 | |
| **Selection**  Selects a subset of the rows within a relation based on a specified condition.  Syntax: *relation* [ *condExpr* ]  Returns a relation which contains rows from the specified relation that meet that conditional expression. | **Example 3:** **Selection -** get enrollment information for students who has a midterm grade >- 90.  Enrollment[Midterm >= 90]   |  |  |  |  | | --- | --- | --- | --- | | **ST#** | **C#** | **Midterm** | **Final** | | 100 | CS 1713 | 95 | 80 | | 100 | MAT 1224 | 100 | 100 | | 200 | CS 1713 | 91 | 90 | | 500 | CS 1713 | 100 | 80 | | 500 | HIS 1033 | 95 | 90 |   **Example 4:** **Selection -** get enrollment information of students who made an A on both exams.  Enrollment[Midterm >= 90 ^ Final >= 90] |
| **Join**  Combines the rows from two relations which satisfy a joining expression. The resulting relation will contain columns from both relations.  Syntax: *relation1* **[** *joinExpr* **]** *relation2*  When the join involves columns of the same name with equivalence, the column can be shown once.  If not using equivalence, show both columns with the relation name as prefixes. | **Example 5:** **Join -** join the Student and Enrollment relations which have the same ST#  Student [ST# = ST#] Enrollment   |  |  |  |  |  | | --- | --- | --- | --- | --- | | **ST#** | **Classif** | **C#** | **Midterm** | **Final** | | 100 | FR | CS 1713 | 95 | 80 | | 100 | FR | MAT 1224 | 100 | 100 | | 100 | FR | ENG 1043 | 85 | 85 | | 100 | FR | HIS 1033 | 80 | 85 | | 200 | SO | CS 1713 | 91 | 90 | | 200 | SO | MAT 1224 | 50 | 80 | | 300 | JR | CS 1713 | 65 | 80 | | 500 | FR | CS 1713 | 100 | 80 | | 500 | FR | HIS 1033 | 95 | 90 | |
| **Division is used for ALL**  Division provides universal quantification.  Let relation **R** (the dividend) have attributes X and Y.  Let relation **S** (the divisor) have attribute Z. Let Y and Z be defined on the same domain.  The divide operation, to divide R by S over Y and Z, syntax is:  R [ Y / Z ] S  The quotient (result) will contain the X-values of R whose corresponding Y-values include **all**  of the values of the divisor S (all of the values of Z).  The dividend should be a binary relation (two columns), the divisor should be a unary relation(one column), and the resulting quotient will be a **unary** relation.  **Notice that we had to put parentheses around the divisor so that it would be done prior to the division.** | **Example 6:** **Division -** get the ST# of students who are enrolled in **all** of the courses.  Enrollment[ST#, C#] [C# / C#] (Course[C#])   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | |  |  | | --- | --- | | **ST#** | **C#** | | 100 | CS 1713 | | 100 | MAT 1224 | | 100 | ENG 1043 | | 100 | HIS 1033 | | 200 | CS 1713 | | 200 | MAT 1224 | | 300 | CS 1713 | | 500 | CS 1713 | | 500 | HIS 1033 | | |  | | --- | | **C#** | | CS 1713 | | MAT 1224 | | ENG 1043 | | HIS 1033 | |   Result:   |  | | --- | | **ST#** | | 100 | |
| **Difference**  This is similar to set difference. This returns the rows from the first relation which are not in the second relation.  Syntax:  *relation1* **-**  *relation2*  The two relations must have the same domains.  **Notice that we had to put parentheses around the subtrahend** (i.e., the operand to the right of the difference operator) **so that it would be done prior to the difference.** | **Example 7: Difference** -Find the ST# of students who **are not** enrolled in "HIS 1033".  Why is this incorrect?  C# not equal to HIS 1033 want ST#  Enrollment [ C# <> "HIS 1033"] [ST#]  Can be in other classes  Incorrect Result:   |  |  |  |  | | --- | --- | --- | --- | | **ST#** | **C#** | **Midterm** | **Final** | | 100 | CS 1713 | 95 | 80 | | 100 | MAT 1224 | 100 | 100 | | 100 | ENG 1043 | 85 | 85 | | 200 | CS 1713 | 91 | 90 | | 200 | MAT 1224 | 50 | 80 | | 300 | CS 1713 | 65 | 80 | | 500 | CS 1713 | 100 | 80 |   Correct answer:  Student [ST#] - (Enrollment [C# = 'HIS 1033'] [ST#] )   |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | Student[ST#]   |  | | --- | | **ST#** | | 100 | | 200 | | 300 | | 400 | | 500 | | Enrollment [C# = 'HIS 1033'] [ST#]   |  | | --- | | **ST#** | | 100 | | 500 | |   Result:   |  | | --- | | **ST#** | | 200 | | 300 | | 400 |   What do you also notice about student 400 based on the data?  Student 400 is not enrolled in any courses |
|  | **Example 8: Difference** - Find the ST# of students who **are only** enrolled in CS 1713.  Student[ST#] – (Enrollment[C# < > “CS 1713”] [ST#])   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | Student[ST#]   |  | | --- | | **ST#** | | 100 | | 200 | | 300 | | 400 | | 500 | | Enrollment (blue for not CS1713 )   |  |  |  |  | | --- | --- | --- | --- | | **ST#** | **C#** | **Midterm** | **Final** | | 100 | CS 1713 | 95 | 80 | | 100 | MAT 1224 | 100 | 100 | | 100 | ENG 1043 | 85 | 85 | | 100 | HIS 1033 | 80 | 85 | | 200 | CS 1713 | 91 | 90 | | 200 | MAT 1224 | 50 | 80 | | 300 | CS 1713 | 65 | 80 | | 500 | CS 1713 | 100 | 80 | | 500 | HIS 1033 | 95 | 90 | |   **Result:**  Student[ST#]   |  | | --- | | **ST#** | | 300 | | 400 |   Note: student 400 actually isn't in any enrollments. Should 400 be in our answer? What can we do to include only students who have enrollments?  Enrollment [ST#] – (Enrollment[C# < > “CS 1713”] [ST#]) |
| **Supplier Parts DB Example**  We will use this classical schema in MANY examples.  SUPPLIER(S#, SNAME, STATUS, CITY)  S# - supplier number  SNAME - supplier name(e.g., SMITH, JONES),  STATUS - a number representing the quality of this supplier with the higher number representing a better quality of service (e.g., 10, 20)  PART(P#, PNAME, COLOR, WEIGHT)  P# - part number  PNAME - part name (e.g., NUT)  COLOR - color of the part (e.g., RED)  WEIGHT - weight of the part in grams  SHIPMENT(S#, P#, QUANTITY)  S# - supplier number  P# - part number  QUANTITY - the number of this part shipped by this supplier  **Constraints**: keys are underlined | **Supplier Parts DB sample Data**   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **SUPPLIER**   |  |  |  |  | | --- | --- | --- | --- | | **S#** | **SNAME** | **STATUS** | **CITY** | | S1 | SMITH | 20 | LONDON | | S2 | JONES | 10 | PARIS | | S3 | BLAKE | 30 | PARIS | | S4 | FORD | 20 | LONDON | | S5 | ADAMS | 30 | ATHENS |   **PART**   |  |  |  |  | | --- | --- | --- | --- | | **P#** | **PNAME** | **COLOR** | **WEIGHT** | | P1 | NUT | RED | 12 | | P2 | BOLT | GREEN | 17 | | P3 | SCREW | BLUE | 17 | | P4 | SCREW | RED | 14 | | P5 | CAM | BLUE | 12 | | P6 | COG | RED | 19 | | **SHIPMENT**   |  |  |  | | --- | --- | --- | | **S#** | **P#** | **QUANTITY** | | S1 | P1 | 300 | | S1 | P2 | 200 | | S1 | P3 | 400 | | S1 | P4 | 200 | | S1 | P5 | 100 | | S1 | P6 | 100 | | S2 | P1 | 300 | | S2 | P2 | 400 | | S3 | P2 | 200 | | S4 | P2 | 200 | | S4 | P4 | 300 | | S4 | P5 | 400 | | |
| We will provide relational algebra queries for problems which we will use with each DML. | **Example SP-1:** Get the part numbers for each of the parts.  PART[P#]  Result:   |  | | --- | | **P#** | | P1 | | P2 | | P3 | | P4 | | P5 | | P6 |   **Example SP-2:**  Get the supplier number, name, and city for each of the suppliers  SUPPLIER [S#, SNAME, CITY]  Result:   |  |  |  | | --- | --- | --- | | S# | SNAME | CITY | | S1 | SMITH | LONDON | | S2 | JONES | PARIS | | S3 | BLAKE | PARIS | | S4 | FORD | LONDON | | S5 | ADAMS | ATHENS | |
|  | **Example SP-3:**  Get the supplier number of suppliers in Paris with a status > 20.  SUPPLIER [CITY = 'PARIS' ^ STATUS > 20] [S#]  Result:   |  | | --- | | S# | | S3 | |
| |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **SUPPLIER**   |  |  |  |  | | --- | --- | --- | --- | | **S#** | **SNAME** | **STATUS** | **CITY** | | S1 | SMITH | 20 | LONDON | | S2 | JONES | 10 | PARIS | | S3 | BLAKE | 30 | PARIS | | S4 | FORD | 20 | LONDON | | S5 | ADAMS | 30 | ATHENS |   **PART**   |  |  |  |  | | --- | --- | --- | --- | | **P#** | **PNAME** | **COLOR** | **WEIGHT** | | P1 | NUT | RED | 12 | | P2 | BOLT | GREEN | 17 | | P3 | SCREW | BLUE | 17 | | P4 | SCREW | RED | 14 | | P5 | CAM | BLUE | 12 | | P6 | COG | RED | 19 | | **SHIPMENT**   |  |  |  | | --- | --- | --- | | **S#** | **P#** | **QUANTITY** | | S1 | P1 | 300 | | S1 | P2 | 200 | | S1 | P3 | 400 | | S1 | P4 | 200 | | S1 | P5 | 100 | | S1 | P6 | 100 | | S2 | P1 | 300 | | S2 | P2 | 400 | | S3 | P2 | 200 | | S4 | P2 | 200 | | S4 | P4 | 300 | | S4 | P5 | 400 | | | **Example SP-4**: Get the supplier number and name of suppliers in London.  SUPPLIER[CITY = ‘LONDON’][S#, SNAME]  **Result:**   |  |  | | --- | --- | | **S#** | **SNAME** | | S1 | SMITH | | S4 | FORD | |
|  | **Example SP-5**: Get the supplier number of suppliers who supply part P2.  SHIPMENT [P# = 'P2'] [S#]   |  | | --- | | **S#** | | S1 | | S2 | | S3 | | S4 | |
| **Join**  Combines the rows from two relations which satisfy a joining expression.  In example SP-6, the table below is the result of the join before the projection of SNAME. Since we joined on S#, we include that column only once.   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **S#** | **P#** | **QUANT ITY** | **SNAME** | **STATUS** | **CITY** | | S1 | P2 | 200 | SMITH | 20 | LONDON | | S2 | P2 | 400 | JONES | 10 | PARIS | | S3 | P2 | 200 | BLAKE | 30 | PARIS | | S4 | P2 | 200 | FORD | 20 | LONDON | | **Example SP-6**: Get the supplier name of suppliers of part P2.  SHIPMENT [P# = 'P2'] [S# = S#] SUPPLIER [SNAME]  Result:   |  | | --- | | SNAME | | SMITH | | JONES | | BLAKE | | FORD |   **Example SP-7:**  Get the part name of parts supplied by supplier S2. (TABLES ABOVE)  SHIPMENT[S# = ‘S2’] [P# = P#] PART [PNAME] |
|  | **Example SP-8:**  Get the city of suppliers who supply part P2. (TABLES ABOVE)  SHIPMENT [P# = ‘P2’] [S# = S#] SUPPLIER [CITY] |
| |  |  |  |  | | --- | --- | --- | --- | | **P#** | **PNAME** | **COLOR** | **WEIGHT** | | P1 | NUT | RED | 12 | | P4 | SCREW | RED | 14 | | P6 | COG | RED | 19 |   **SHIPMENT**   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **S#** | **P#** | **QUANT ITY** | **PNAME** | **COLOR** | **WEIGHT** | | S1 | P1 | 300 | NUT | RED | 12 | | S1 | P4 | 200 | SCREW | RED | 14 | | S1 | P6 | 100 | COG | RED | 19 | | S2 | P1 | 300 | NUT | RED | 12 | | S4 | P4 | 300 | SCREW | RED | 14 | | **Example SP-9:**  Get the supplier number of suppliers who supply at least one red part.  PART[COLOR = 'RED'] [P# = P#] SHIPMENT [S#]  Result:   |  | | --- | | S# | | S1 | | S2 | | S4 | |
|  | **Example SP-10:**  Get the name of suppliers who supply at least one red part.  PART[COLOR = 'RED'][P# = P#] SHIPMENT[S# = S#] SUPPLIER [SNAME] |
| Some queries require multiple references to the same relation. In SP-11, we need to use SHIPMENT twice:  1. Select the parts supplied by supplier S2.  **SHIPMENT by supplier S2**   |  |  |  | | --- | --- | --- | | **S#** | **P#** | **QUANTITY** | | S2 | P1 | 300 | | S2 | P2 | 400 |   2. For the P# from those shipments, we need to join those back with SHIPMENT to obtain other suppliers of at least one of those two parts. The rows in blue match those parts and would be in the results. (The other rows are simply shown to see what was in the SHIPMENT relation.)  **SHIPMENT**   |  |  |  | | --- | --- | --- | | **S#** | **P#** | **QUANTITY** | | S1 | P1 | 300 | | S1 | P2 | 200 | | S1 | P3 | 400 | | S1 | P4 | 200 | | S1 | P5 | 100 | | S1 | P6 | 100 | | S2 | P1 | 300 | | S2 | P2 | 400 | | S3 | P2 | 200 | | S4 | P2 | 200 | | S4 | P4 | 300 | | S4 | P5 | 400 | | **Example SP-11:**  Get the S# and names of suppliers who supply at least one part supplied by supplier S2.  SHIPMENT[S# = 'S2'] **[P#]** [P# = P#] SHIPMENT  [S#=S#] SUPPLIER [S#, SNAME]  Why should we project P#? We have an S# in the first SHIPMENT. If we keep it in there, when we join the other SHIPMENT, we will have two S# columns. We said the join on P# would result in just one P# column, but without that projection of P#, we would have two S# columns. Instead of dealing with an additional syntax issue, we can simply remove the first S# column by projecting P#.  Will our result include S2? yes, since it also supplies at least one of those parts.   |  |  | | --- | --- | | **S#** | **SNAME** | | S1 | SMITH | | S2 | JONES | | S3 | BLAKE | | S4 | FORD |   How would we remove S2 from the result?  SHIPMENT[S# = 'S2'] **[P#]** [P# = P#] SHIPMENT  [S#=S#] SUPPLIER [S# <> 'S2'] [S#, SNAME] |
| **When taking the difference, we must be certain that each of the operands is representing the correct information. It isn't simply an issue of subtracting the same columns.**  Notice that A and B are subtracting from different relations.   |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | A. From SUPPLIER  **SUPPLIER**   |  | | --- | | **S#** | | S1 | | S2 | | S3 | | S4 | | S5 | | B. From SHIPMENT  **SHIPMENT**   |  | | --- | | **S#** | | S1 | | S2 | | S3 | | S4 | | | **Example SP-12:**  Get the supplier names of suppliers who do **not** supply part P1.  **Which of these queries are correct? What are the incorrect ones doing?**  **A.** SUPPLIER[S#] - (SHIPMENT[P# = 'P1'] [S#] )  [S# = S#] SUPPLIER [SNAME]  **B.** SHIPMENT[S#] - (SHIPMENT[P# = 'P1'] [S#] )  [S# = S#] SUPPLIER [SNAME]  **C.** SHIPMENT- (SHIPMENT[P# = 'P1'] )  [S# = S#] SUPPLIER [SNAME]  Both A and B are subtracting suppliers (via S#). Answer A includes suppliers that do not have any shipments. Answer B only includes suppliers who supply something. Based on the request A is a better answer.  **C is incorrect. Why?**  S1 would be in our answers  **For C: (subtracting only the rows in blue)**  **SHIPMENT**   |  |  |  | | --- | --- | --- | | **S#** | **P#** | **QUANTITY** | | S1 | P1 | 300 | | S1 | P2 | 200 | | S1 | P3 | 400 | | S1 | P4 | 200 | | S1 | P5 | 100 | | S1 | P6 | 100 | | S2 | P1 | 300 | | S2 | P2 | 400 | | S3 | P2 | 200 | | S4 | P2 | 200 | | S4 | P4 | 300 | | S4 | P5 | 400 | |
| **ALL** | **Example SP-13:**  Get the names of suppliers who supply **all** of the parts.  SHIPMENT[S#,P#] [P# / P#] (PART[P#])  [S# = S#] SUPPLIER [SNAME] |
| This is the same as **SP-13,** but we already have **P#** so we don’t have to join with **SUPPLIER.** | **Example SP-14:**  Get the supplier numbers of suppliers who supply all the parts.  SHIPMENT[S#,P#] [P# / P#] (PART[P#]) |
| **ONLY**  We must subtract out the opposite. In this example we must subtract out suppliers who supply a part other than P2.   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **SUPPLIER**   |  |  |  |  | | --- | --- | --- | --- | | **S#** | **SNAME** | **STATUS** | **CITY** | | S1 | SMITH | 20 | LONDON | | S2 | JONES | 10 | PARIS | | S3 | BLAKE | 30 | PARIS | | S4 | FORD | 20 | LONDON | | S5 | ADAMS | 30 | ATHENS |   **PART**   |  |  |  |  | | --- | --- | --- | --- | | **P#** | **PNAME** | **COLOR** | **WEIGHT** | | P1 | NUT | RED | 12 | | P2 | BOLT | GREEN | 17 | | P3 | SCREW | BLUE | 17 | | P4 | SCREW | RED | 14 | | P5 | CAM | BLUE | 12 | | P6 | COG | RED | 19 | | **SHIPMENT**   |  |  |  | | --- | --- | --- | | **S#** | **P#** | **QUANTITY** | | S1 | P1 | 300 | | S1 | P2 | 200 | | S1 | P3 | 400 | | S1 | P4 | 200 | | S1 | P5 | 100 | | S1 | P6 | 100 | | S2 | P1 | 300 | | S2 | P2 | 400 | | S3 | P2 | 200 | | S4 | P2 | 200 | | S4 | P4 | 300 | | S4 | P5 | 400 | | | **Example SP-15:** Get the names of suppliers who **only** supply part P2.  What is different in these results?  <> == != or something other  Ex) Base – (something)[join]exGroup[projection]  A. SHIPMENT[S#] - (SHIPMENT[P# <> 'P2'][S#])  [S# = S#] SUPPLIER [SNAME]  B. SUPPLIER[S#] - (SHIPMENT[P# <> 'P2'][S#])  [S# = S#] SUPPLIER [SNAME]  With B, you get suppliers in the result that didn't ship anything. |
| **ALL of a subset (selection then division)**   * First decide what is it that we want all of * determine the subset via selection * divide by what we want all of   Notice that it is **necessary to place parentheses** around the selection of read parts and projected P#. This causes that to be done prior to the division. | **Example SP-16:** Get the names of suppliers who supply **all** of the red parts.  How do we get the P# of the red parts?  PART[COLOR = 'RED'] [P#]  **Query:**  SHIPMENT[S#,P#] [P# / P#] (PART[COLOR='RED'][P#])  [S# = S#] SUPPLIER [SNAME] |
| **ONLY**  We must subtract out the opposite.   * Determine what we are subtracting. We are subtracting parts. * Subtract out parts supplied by suppliers not in London | **Example SP-17:** Get the name and part number of parts **only** supplied by suppliers in London.  How do we get parts supplied by suppliers not in London? (need just P#)  SUPPLIER[CITY <> 'LONDON'] [S#=S#] SHIPMENT [P#]  **Supplier Parts DB sample Data**   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **SUPPLIER**   |  |  |  |  | | --- | --- | --- | --- | | **S#** | **SNAME** | **STATUS** | **CITY** | | S1 | SMITH | 20 | LONDON | | S2 | JONES | 10 | PARIS | | S3 | BLAKE | 30 | PARIS | | S4 | FORD | 20 | LONDON | | S5 | ADAMS | 30 | ATHENS |   **Parts supplied by suppliers not in London**   |  | | --- | | **P#** | | P1 | | P2 | | P3 | | P4 | | P5 | | P6 | | **SHIPMENT**   |  |  |  | | --- | --- | --- | | **S#** | **P#** | **QUANTITY** | | S1 | P1 | 300 | | S1 | P2 | 200 | | S1 | P3 | 400 | | S1 | P4 | 200 | | S1 | P5 | 100 | | S1 | P6 | 100 | | S2 | P1 | 300 | | S2 | P2 | 400 | | S3 | P2 | 200 | | S4 | P2 | 200 | | S4 | P4 | 300 | | S4 | P5 | 400 | | |
|  | **Example SP17 continued:** Get the name and part number of parts **only** supplied by suppliers in London.  We know which parts not to include so we subtract those from PART  PART[P#] - (SUPPLIER[CITY <> 'LONDON'] [S#=S#] SHIPMENT [P#])  [p#=P#] PART [P#, PNAME]  **Result:**   |  |  | | --- | --- | | **P#** | **PNAME** | | P3 | SCREW | | P4 | SCREW | | P5 | CAM | | P6 | COG | |
| **Selection from multiple relations with multiple criteria upplier Parts**  **DB sample Data**   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **SUPPLIER**   |  |  |  |  | | --- | --- | --- | --- | | **S#** | **SNAME** | **STATUS** | **CITY** | | S1 | SMITH | 20 | LONDON | | S2 | JONES | 10 | PARIS | | S3 | BLAKE | 30 | PARIS | | S4 | FORD | 20 | LONDON | | S5 | ADAMS | 30 | ATHENS |   **Parts supplied by suppliers not in London**   |  | | --- | | **P#** | | P1 | | P2 | | P3 | | P4 | | P5 | | P6 | | **SHIPMENT**   |  |  |  | | --- | --- | --- | | **S#** | **P#** | **QUANTITY** | | S1 | P1 | 300 | | S1 | P2 | 200 | | S1 | P3 | 400 | | S1 | P4 | 200 | | S1 | P5 | 100 | | S1 | P6 | 100 | | S2 | P1 | 300 | | S2 | P2 | 400 | | S3 | P2 | 200 | | S4 | P2 | 200 | | S4 | P4 | 300 | | S4 | P5 | 400 | | | **Example SP18:** For shipments with a quantity < 500 by suppliers with a status greater than 10 of a part with a weight greater than 12, get the part#. **FINAL LEVEL AND GR EXAMPLE**  SUPPLIER [STATUS > 10]  [S#=S#]  SHIPMENT[QUANTITY < 500]  [P#=P#]  PART [WEIGHT > 12]  Result:   |  | | --- | | P# | | P2 | | P3 | | P4 | | P6 |   **PART [WEIGHT > 12]**  **[P#=P#]**  **SHIPMENT[QUANTITY<500]**  **[S# = S#]**  **SUPPLIER[STATUS > 10]** |
| **Selection from multiple relations with multiple criteria**   |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **SUPPLIER**   |  |  |  |  | | --- | --- | --- | --- | | **S#** | **SNAME** | **STATUS** | **CITY** | | S1 | SMITH | 20 | LONDON | | S2 | JONES | 10 | PARIS | | S3 | BLAKE | 30 | PARIS | | S4 | FORD | 20 | LONDON | | S5 | ADAMS | 30 | ATHENS |   **Parts supplied by suppliers not in London**   |  | | --- | | **P#** | | P1 | | P2 | | P3 | | P4 | | P5 | | P6 | | **SHIPMENT**   |  |  |  | | --- | --- | --- | | **S#** | **P#** | **QUANTITY** | | S1 | P1 | 300 | | S1 | P2 | 200 | | S1 | P3 | 400 | | S1 | P4 | 200 | | S1 | P5 | 100 | | S1 | P6 | 100 | | S2 | P1 | 300 | | S2 | P2 | 400 | | S3 | P2 | 200 | | S4 | P2 | 200 | | S4 | P4 | 300 | | S4 | P5 | 400 | | | **Example SP19:** For shipments with a quantity < 500 by suppliers with a status greater than 10 of a part with a weight greater than 12, get the part number, part name, supplier number, and supplier name.  SUPPLIER [STATUS > 10]  [S#=S#]  SHIPMENT[QUANTITY < 500]  [P#=P#]  PART [WEIGHT > 12] [P#, PNAME, S#, SNAME]  Result:   |  |  |  |  | | --- | --- | --- | --- | | P# | PNAME | S# | SNAME | | P2 | BOLT | S1 | SMITH | | P2 | BOLT | S3 | BLAKE | | P2 | BOLT | S4 | FORD | | P3 | SCREW | S1 | SMITH | | P4 | SCREW | S1 | SMITH | | P4 | SCREW | S4 | FORD | | P6 | COG | S1 | SMITH | |
|  |  |
|  |  |

**©2020 Larry W. Clark,** UTSA CS students may make copies for their personal use