# Relational Algebra Equivalences

1. All joins and products are commutative.

- R x S = S x R
- $R \bowtie_{\theta} S = S \bowtie_{\theta} R$
- $R \bowtie S = S \bowtie R$

2. Joins and products are associative

- (R x S) x T = R x (S x T)
- $(R \bowtie_{\theta} S) \bowtie_{\theta} T = R \bowtie_{\theta} (S \bowtie_{\theta} T)$
- $(R \bowtie S) \bowtie T = R \bowtie (S \bowtie T)$

3. Select is commutative

- $\sigma_p ( \sigma_q (R)) = \sigma_q (\sigma_p (R))$

4. Conjunctive selects can cascade into individual selects

- $\sigma_{p \&q\&\ldots\&z} (R) = (\sigma_p (\sigma_q ...( \sigma_z (R))...))$

5. Successive projects can reduced to the final project.

- If $list_1$, $list_2$, … $list_n$ are lists of attribute names and each of the $list_i$ contains $list_{i-1}$,
  then $\Pi_{list1} (\Pi_{list2} (...\Pi_{listn} (R)...) = \Pi_{list1}(R)$
- So only the last project has to be executed

6. Select and project sometimes commute

- If p involves only the attributes in projlist, then select and project commute
- $\Pi_{projlist} ( \sigma_p (R)) = \sigma_p ( \Pi_{projlist} (R))$

7. Select and join (or product) sometimes commute

- If p involves only attributes of one of the tables being joined, then select and join commute
  $\sigma_p (R \bowtie S) = (\sigma_p (R)) \bowtie S$
- Only if p refers just to R

8. Select sometimes distributes over join (or product)

- For p AND q, where p involves only the attributes of R and q only the attributes of S the select distributes over the join
- $\sigma_{p \text{ AND } q} (R \bowtie S) = ( \sigma_p (R)) \bowtie ( \sigma_q (S))$

9. Project sometimes distributes over join (or product)

- If projlist can be split into separate lists, list1 and list2, so that list1 contains only attributes of R and list2 contains only attributes of S, then
- $\Pi_{projlist} (R \bowtie S) = ( \Pi_{list1} (R)) \bowtie ( \Pi_{list2} (S))$

10. Union and intersection are commutative

- $R \cup S = S \cup R$
- $R \cap S = S \cap R$
- Set difference is not commutative.

11. Union and intersection are individually associative.

- $(R \cup S) \cup T = R \cup (S \cup T)$
- $(R \cap S) \cap T = R \cap (S \cap T)$
- Set difference is not associative

12. Select distributes over union, intersection, and difference

- $\sigma_p (R \cup S) = \sigma_p (R) \cup \sigma_p (S)$
- $\sigma_p (R \cap S) = \sigma_p (R) \cap \sigma_p (S)$
- $\sigma_p (R - S) = \sigma_p (R) - \sigma_p (S)$

13. Project distributes over union, intersection, and difference

- $\Pi_{projlist} (R \cup S) = (\Pi_{projlist} (R)) \cup (\Pi_{projlist} (S))$
- $\Pi_{projlist} (R \cap S) = (\Pi_{projlist} (R)) \cap (\Pi_{projlist} (S))$
- $\Pi_{projlist} (R - S) = (\Pi_{projlist} (R)) - (\Pi_{projlist} (S))$

14. Project is idempotent-repeating it produces the same result

- $\Pi_{projlist} (R)( \Pi`_{projlist}(R)) = \Pi_{projlist}R)$

15. Select is idempotent

- $\sigma_p(\sigma_p(R)) = \sigma_p(R)$