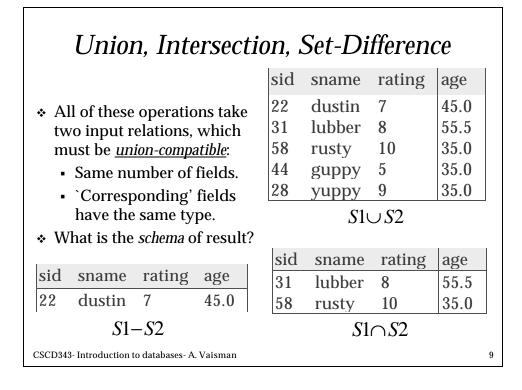


| Example Instanc | es | R 1 | <u>sid</u> 22 58 | <u>bio</u> 10 10 | 01 10/ | ay 10/96 12/96 | - |
|--|-----------|----------------|--------------------------|------------------------|--------------|----------------------|---|
| relations for our examples. | S1 | sid | snam | | rating | age | |
| "bid" = boats. "sid": sailors We'll use positional or named field notation, assume that names of fields in query results are `inherited' from names of fields in query input relations. | | 22 31 58 | dustin lubbe rusty | | 7 8 10 | 45.0 55.5 35.0 | |
| | <i>S2</i> | sid | snam | e | rating | age | |
| | | 28 31 | yupp lubbe | - | 9 8 | 35.0 55.5 | |
| | | 44 58 | gupp rusty | У | 5 10 | 35.0 35.0 | |
| CSCD343- Introduction to databases- A. Vaisman | | | | | | | 5 |

Relational Algebra Basic operations: Selection (*S*) Selects a subset of rows from relation. *Projection* (*P*) Deletes unwanted columns from relation. *Cross-product* (×) Allows us to combine two relations. *Set-difference* (−) Tuples in reln. 1, but not in reln. 2. *Union* (∪) Tuples in reln. 1 and in reln. 2. Additional operations: Intersection, *join*, division, renaming: Not essential, but (very!) useful. Since each operation returns a relation, operations can be composed! (Algebra is "closed".)

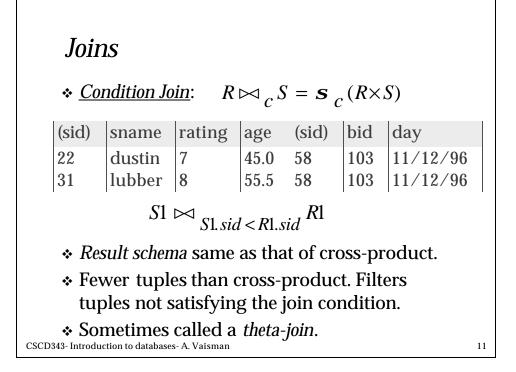
| Projection operator has to eliminate <i>duplicates</i>! (Why??, what are the consequences?) Note: real systems typically don't do duplicate elimination unless the user explicitly asks for it. (Why not?) age 35.0 55.5 <i>p_{age}(S2)</i> | Projection Deletes attributes that are not in projection list. Schema of result contains exactly the fields in the projection list, with the same names that they had in the (only) input relation. | yuppy lubber guppy rusty | rating 9 8 5 10 rating ^(S2) | |
|--|---|-----------------------------------|---|---|
| don't do duplicate elimination unless the user explicitly asks for it. (Why not?) $P_{age}(S2)$ | eliminate <i>duplicates</i> ! (Why??, | | | |
| CSCD345 Infoduction to databases A: valsman | don't do duplicate elimination unless the user explicitly asks | | | 7 |

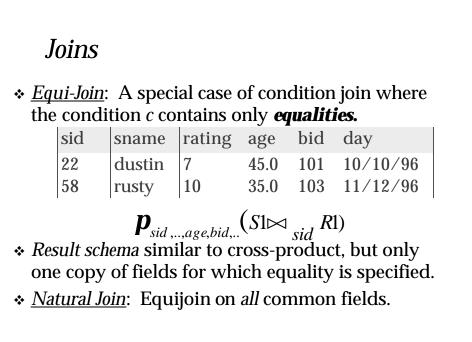
| Selection | sid 28 58 | sname yuppy rusty | rating 9 10 | age 35.0 35.0 | |
|--|------------------------|---------------------------------------|---|----------------------|---|
| Selects rows that satisfy selection condition. Schema of result identical to schema of (only) input relation. | | | ing>8 ^{(SZ} | • | |
| <i>Result</i> relation can be the <i>input</i> for another relational algebra operation! (<i>Operator</i> <i>composition</i>.) | p _{sv} | sname yuppy rusty aame,ratin | rating 9 10 g(S rating | g>8 ^(S2)) | |
| CSCD343- Introduction to databases- A. Vaisman | 1 | | | | 8 |



Cross-Product ✤ Each row of S1 is paired with each row of R1. ✤ Result schema has one field per field of S1 and R1, with field names `inherited' if possible. • *Conflict*: Both S1 and R1 have a field called *sid*. (sid) sname rating age (sid) bid day 22 dustin 7 45.0 22 101 10/10/96 22 dustin 7 45.0 58 103 11/12/96 31 lubber 8 55.5 22 101 10/10/96 31 lubber 8 55.5 58 103 11/12/96 10 35.0 22 101 10/10/96 58 rusty 10 35.0 11/12/96 58 58 103 rusty r ($C(1 \rightarrow sid1, 5 \rightarrow sid2), S1 \times R1$) Renaming operator: CSCD343- Introduction to databases- A. Vaisman

10





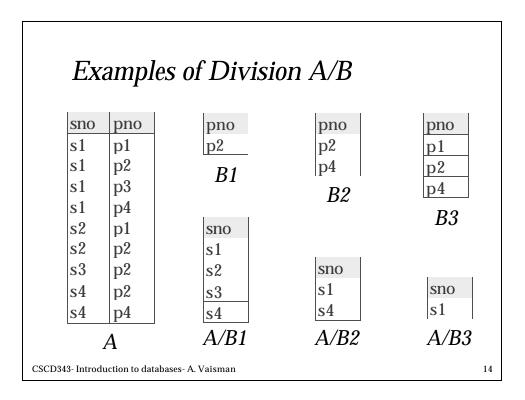
Division

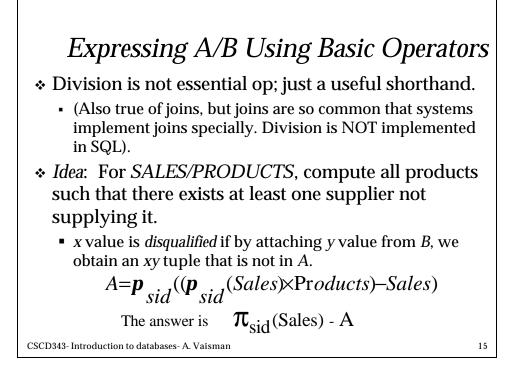
 Not supported as a primitive operator, but useful for expressing queries like:

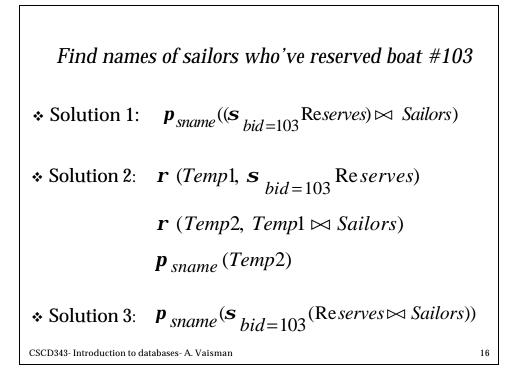
Find sailors who have reserved **<u>all</u>** boats.

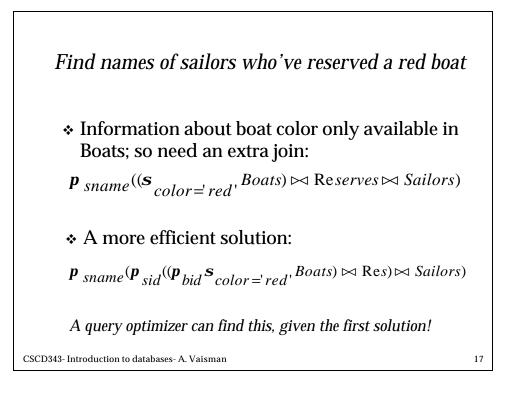
- Precondition: in A/B, the attributes in B must be included in the schema for A. Also, the result has attributes A-B.
 - SALES(supId, prodId);
 - PRODUCTS(prodId);
 - Relations SALES and PRODUCTS must be built using projections.
- SALES/PRODUCTS: the ids of the suppliers supplying CSCD343Andrduproductsses- A. Vaisman

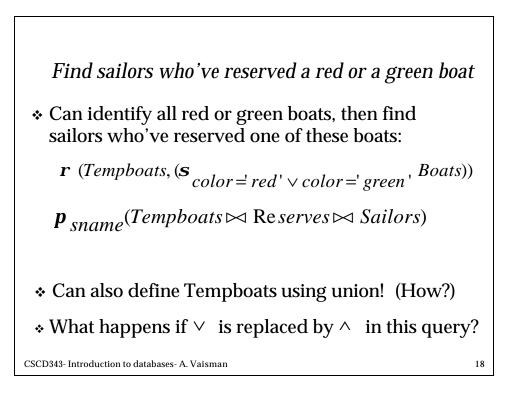
13

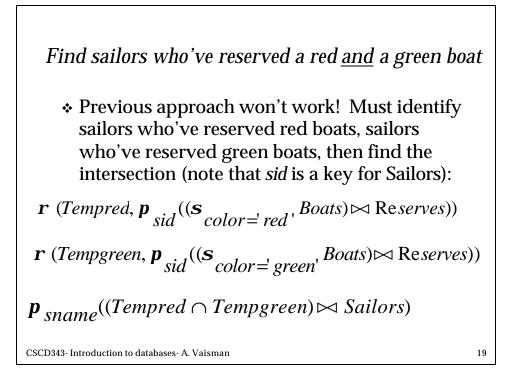


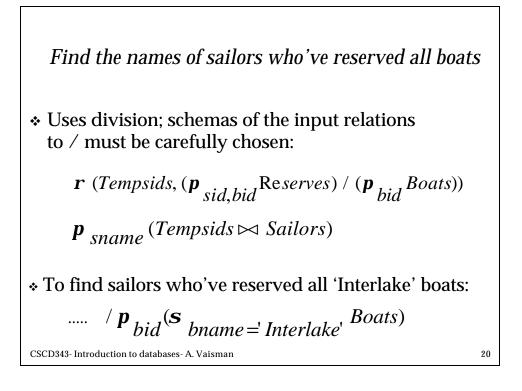


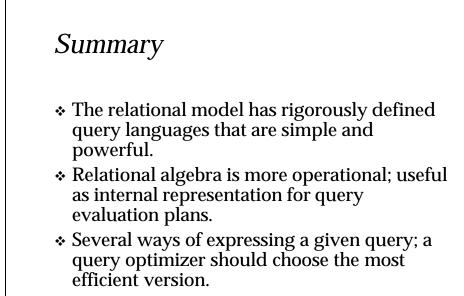












CSCD343- Introduction to databases- A. Vaisman

21