Relational Calculus

- Queries are open predicate-calculus formulas
- General form: \( \{ <x_1, \ldots, x_n> \mid F(x_1, \ldots, x_n) \} \)
  - Standard set-builder notation (i.e. constructs a set of tuples)
  - \( x_1, \ldots, x_n \) are free variables
  - \( F \) is a predicate calculus formula
    - Bound variables as well as free variables
    - Built-in predicates, e.g. \(<, =, \ldots\>
    - Relation predicates, e.g. \( r(x, y, 2, z) \)
    - Standard Boolean connectors, e.g. \( \neg, \land, \lor, \ldots \)
    - Standard quantifiers: \( \exists \) and \( \forall \)
Relational Calculus
Basic Project-Select-Join
Examples

Get room information.

\{ \langle x, y, z, w \rangle \mid r(x, y, z, w) \}\n
Get room number and room name of rooms that cost less than $85 and have 2 beds.

\{ \langle x, y \rangle \mid \exists z(r(x, y, 2, z) \land z < 85) \}\n
Get name and address of guests arriving on 10 May.

\{ \langle x, y, z \rangle \mid \exists u \exists v \exists w(g(u, x, y, z) \\
\land s(u, v, 10 \text{ May}, w)) \}\
Relational Calculus
Join, Renaming & Union
Examples

Get name and address of guests who have a Reservation for a room whose name is the same as the guest's name.

\{ <x, y, z> | \exists a \exists b \exists c \exists d \exists e \exists f(g(a, x, y, z) \\
\land r(b, x, c, d) \land s(a, b, e, f)) \} \\

Get name and address of guests who have reservations for more than two days or reservations for two-bed rooms.

\{ <x, y:StreetAddr, z:Location> | \\
\exists a \exists b \exists c \exists d \exists e \exists f(g(a, x, y, z) \\
\land s(a, b, c, d) \land (d > 2 \lor r(b, e, 2, f))) \}
Relational Calculus
Negation

Get guest number and name of guests not from Boston.

\{ <x, y> \mid \exists z \exists w (g(x, y, z, w) \land w \neq \text{Boston}) \} 

Get guest number of guests who do not have a reservation for room 1.

The following is not correct.

\{ <x> \mid \exists y \exists z \exists w (s(x, y, z, w) \land y \neq 1) \}
Relational Calculus
Negation and Universal
Quantification

Get guest number of guests who do not have a Reservation for room 1. (continuation of example)

Find those who do and negate.

\[
\{ <x> \mid \neg \exists y \exists z \exists w (s(x, y, z, w) \land y = 1) \} \\
= \{ <x> \mid \forall y \forall z \forall w \neg (s(x, y, z, w) \land y = 1) \} \\
= \{ <x> \mid \forall y \forall z \forall w (\neg s(x, y, z, w) \lor y \neq 1) \}
\]

Almost correct, but yields universal complement. Restrict by using relative complement.

\[
\{ <x> \mid \exists t \exists u \exists v (g(x, t, u, v) \\
\quad \land \neg \exists y \exists z \exists w (s(x, y, z, w) \land y = 1)) \}
\]
Relational Calculus
Universal Quantification

Get name and address of guests who have reservations for all presidential suites (rooms with two beds).

\{ <x, y, z> | \exists w (g(w, x, y, z) \land \\
\forall a \forall b \forall c (r(a, b, 2, c) \\
\rightarrow \exists d \exists e (s(w, a, d, e)))) \}
Universal Quantification and SQL Queries

\{ <x, y, z> | \exists w(g(w, x, y, z) \land \\
\forall a \forall b \forall c(r(a, b, 2, c) \rightarrow \exists d \exists e(s(w, a, d, e)))) \}\}

= \{ <x, y, z> | \exists w(g(w, x, y, z) \land \\
\neg \forall a \forall b \forall c(\neg r(a, b, 2, c) \\
\lor \exists d \exists e(s(w, a, d, e)))) \}\}

= \{ <x, y, z> | \exists w(g(w, x, y, z) \land \\
\neg \exists a \exists b \exists c(r(a, b, 2, c) \land \\
\neg \exists d \exists e(s(w, a, d, e)))) \}\}

= \text{select Name, StreetNr, City} \\
\text{from Guest g where} \\
\text{not exists (select * from Room r} \\
\text{where NrBeds = 2 and} \\
\text{not exists (select *} \\
\text{from Reservation s where} \\
g.GuestNr = s.GuestNr \\
\text{and r.RoomNr = s.RoomNr))}

RelCalc: 7