MBDA 01

LABORATORIO 01/02/19

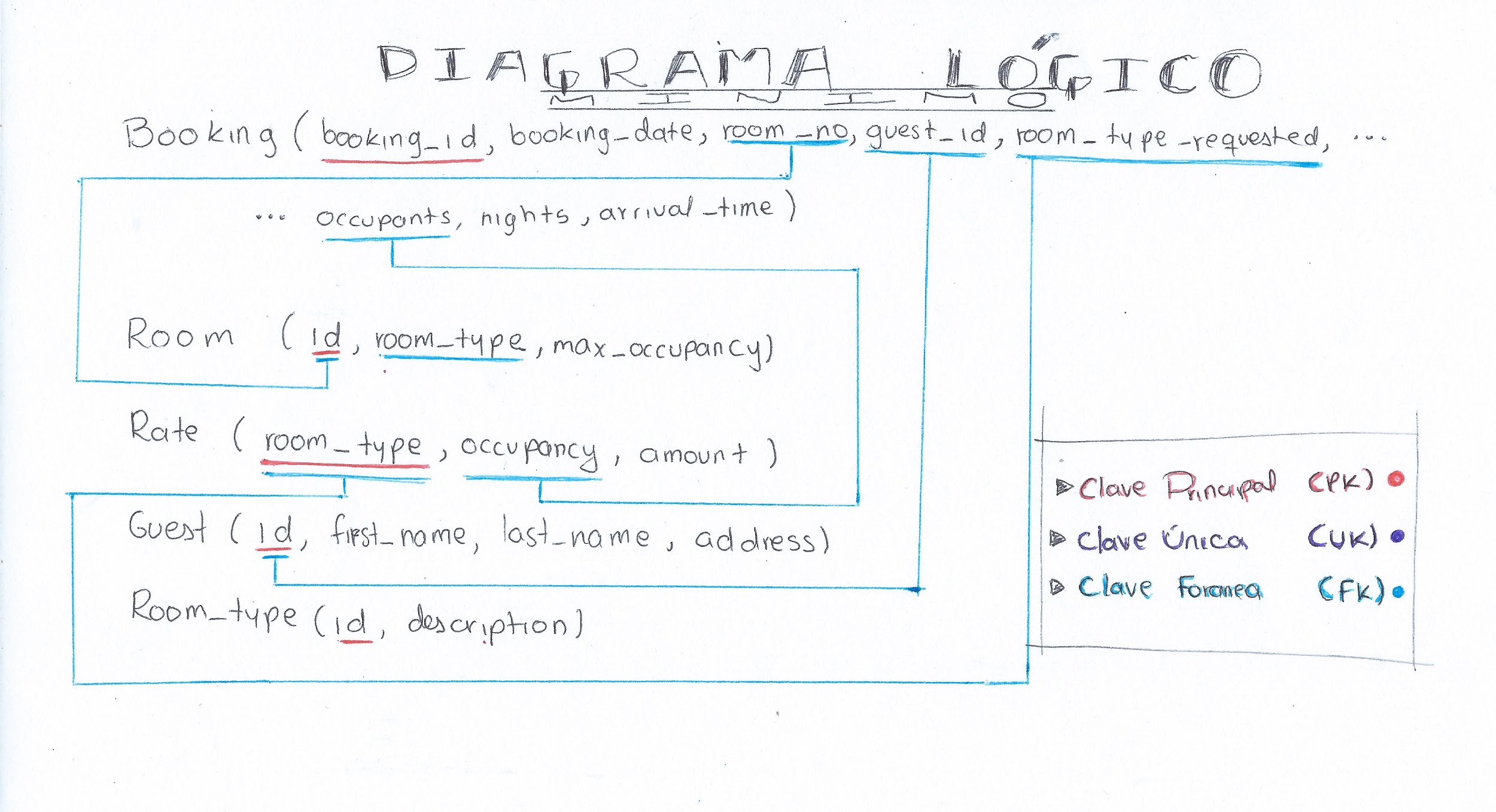
Línea horizontal

# AUTORES

Laura Izquierdo, Yarit Villalobos

# PARTE UNO. Conociendo la organización A. Ingeniería reversa

1. Diagrama Lógico mínimo



2. Diagrama de conceptos sin atributos

Ver Archivo anexo en **guest.asta → (Concepts Class Diagram)**

# 

# B. Revisando el contenido

# 1. ¿Qué tipos de cuartos ofrecen?

SELECT DISTINCT room\_type AS 'TipoCuarto'  
FROM room

# 2. ¿Cuántos habitaciones tiene? ¿Cuáles es su capacidad total?

SELECT COUNT(id) AS 'No\_rooms',SUM(max\_occupancy) AS 'TotalCapacity'

FROM room

# 3. ¿Cuántos clientes tiene?¿Cuántos todavía no han hecho reservas?

SELECT COUNT(id) AS 'Cliente',COUNT(id)-(SELECT COUNT(guest\_id) FROM booking ) AS 'SinReserva'

FROM guest

# 4. ¿Cuántas reservas tiene actualmente? ¿Entre que fechas?

# SELECT COUNT(booking\_date),MIN(booking\_date),MAX(booking\_date)

# FROM booking

# 5. Proponer una pregunta y responderla

# ¿Cuál es la tarifa mínima de un cuarto?¿De qué tipo es el mínimo?

SELECT room\_type,amount

FROM rate

WHERE amount LIKE (SELECT min(amount) FROM rate)

# C. Contexto

# 1. Misión. ¿Cuál creen que es la misión de la organización?

Satisfacer las necesidades de los huéspedes, completando la estadía con la atención y servicios.

# 2. Servicios. ¿Qué ofrece a sus clientes?

La mejor atención en la reservación de los cuartos, desayuno al cuarto, minibar.

D. Usuario

# ¿Cuáles son tres posibles usuarios de esta información? ¿Qué papel juegan en la organización?

* **Interno:** Por ejemplo, el administrador del hotel.
* **Externo:** Por ejemplo, una plataforma de planes turísticos. Promueve el uso de la organización en cuestión.
* **Cliente:** Por ejemplo, el huésped.

Adicionalmente, ver archivo anexo en **guest.asta (UseCase Diagram Part\_D)**

# PARTE DOS. Implementando

# Easy questions

**1.Give the booking\_date and the number of nights for guest 1183.**

**Álgebra:** Π booking\_date,nights **σ** guest\_id = 1183 booking

**Cálculo:** {x : booking| x.guest\_id = '1183': x.booking\_date,x.nights}

**SQL:**

SELECT booking\_date,nights

FROM booking

WHERE guest\_id = '1183'

2.**List the arrival time and the first and last names for all guests due to arrive on 2016-11-05, order the output by time of arrival.**

**Álgebra:** Π arrival\_time,first\_name,last\_name **σ** booking\_date = 2016-11-05 booking,guest

**Cálculo:** {x : booking, y : guest| x.booking\_date= 2016-11-05: x.arrival\_time,y.first\_name,y.last\_name}

**SQL:**

SELECT arrival\_time, first\_name,last\_name

FROM booking JOIN guest ON guest\_id = guest.id

WHERE booking\_date= '2016-11-05'

ORDER BY arrival\_time asc

**3.Give the daily rate that should be paid for bookings with ids 5152, 5165, 5154 and 5295. Include booking id, room type, number of occupants and the amount.**

**Álgebra:** Π booking\_id,max(room\_type\_requested),max(occupants),max(amount) **σ**booking\_id = 5152,booking\_id = 5154,booking\_id = 5165,booking\_id = 5295 booking,rate

**Cálculo:** {x : booking| x.booking\_id = 5152,x.booking\_id = 5154,x.booking\_id = 5165,x.booking\_id = 5295: x.booking\_id,max(x.room\_type\_requested),max(x.occupants),max(x.amount)}

**SQL:**

SELECT booking\_id,max(room\_type\_requested)AS 'room\_type\_requested',max(occupants)AS 'occupants',max(amount) AS 'amount'

FROM booking JOIN rate ON occupants=rate.occupancy

WHERE booking\_id IN ('5152','5154','5165','5295')

GROUP BY booking\_id

**4.Find who is staying in room 101 on 2016-12-03, include first name, last name and address.**

**Álgebra:** Π first\_name,last\_name,address **σ** room\_no=101,booking\_date=2016-12-03 booking,guest

**Cálculo:** {x : booking, y : guest| x.room\_no=101,x.booking\_date=2016-12-03:y.first\_name,y.last\_name,y.address}

**SQL:**

SELECT first\_name,last\_name,address

FROM booking JOIN guest ON guest\_id = guest.id

WHERE room\_no=101 AND booking\_date='2016-12-03'

**5.The guest id and the total number of bookings and the total number of nights.**

**Álgebra:** Π guest\_id,count(booking\_id),sum(nights) **σ**guest\_id = 1185, guest\_id = 1270 booking

**Cálculo:** {x : booking| x.guest\_id = 1185, x.guest\_id = 1270: x.guest\_id,count(x.booking\_id),sum(x.nights)}

**SQL:**

SELECT guest\_id,COUNT(booking\_id),SUM(nights)

FROM booking

WHERE guest\_id IN('1185','1270')

GROUP BY guest\_id

# Medium questions

**6.Ruth Cadbury. Show the total amount payable by guest Ruth Cadbury for her room bookings.**

**Cálculo:** {x : booking, y: rate, z: guest| y.occupancy=x.occupants, y.room\_type=x.room\_type\_requested,x.guest\_id=z.id,z.first\_name=Ruth, z.last\_name=Cadbury: sum(x.nights\*y.amount)}

**SQL:**

SELECT SUM(nights\*amount)  
FROM booking JOIN rate ON (occupancy=occupants AND room\_type=room\_type\_requested)  
WHERE guest\_id= (SELECT id FROM guest WHERE first\_name='Ruth' and last\_name='Cadbury')

**7.Including Extras. Calculate the total bill for booking 5346 including extras**

**Cálculo:** {x : booking, y: rate, z: extra| y.occupancy=x.occupants, y.room\_type=x.room\_type\_requested,x.booking\_id=z.booking\_id,x.booking\_id='5346': y.amount+SUM(z.amount)}

**SQL:**

SELECT rate.amount+SUM(extra.amount) AS 'SUM(amount)'

FROM booking

JOIN rate ON (occupancy=occupants AND room\_type=room\_type\_requested)

JOIN extra ON booking.booking\_id=extra.booking\_id

WHERE booking.booking\_id='5346'

GROUP BY rate.amount

**8.Edinburgh Residents. For every guest who has the word “Edinburgh” in their address show the total number of nights booked. Be sure to include 0 for those guests who have never had a booking.**

**Cálculo:** {x : guest, y: booking| x.id=y.guest\_id, x.address='%Edinburgh%',IFNULL(SUM(y.nights))THEN 0: x.last\_name,x.first\_name,x.address,SUM(y.nights)}

**SQL:**

SELECT last\_name, first\_name, address, IFNULL(sum(nights), 0) as nights

FROM guest

LEFT OUTER JOIN booking ON booking.guest\_id = guest.id

WHERE address LIKE '%Edinburgh%'

GROUP BY last\_name,first\_name,address

**9.How busy are we? For each day of the week beginning 2016-11-25 show the number of bookings starting that day.**

**Cálculo:** {x : booking| x.booking\_date>=2016-11-25, x.booking\_date<=2016-12-01: x.booking\_date,COUNT(x.booking\_date)}

**SQL:**

SELECT booking\_date AS i,COUNT(booking\_date) AS arrivals

FROM booking

WHERE booking\_date BETWEEN '2016-11-25' AND '2016-12-01'

GROUP BY booking\_date

**10.How many guests? Show the number of guests in the hotel on the night of 2016-11-21.**

**Cálculo:** {x : booking| x.booking\_date<=2016-11-21, (x.booking\_date + INTERVAL x.nights DAY) > 2016-11-21: SUM(x.occupants)}

**SQL:**

SELECT SUM(occupants)

FROM booking

WHERE booking\_date <='2016-11-21' AND (booking\_date + INTERVAL nights DAY )> '2016-11-21'

# Hard questions

**12.Check out per floor. The first digit of the room number indicates the floor – e.g. room 201 is on the 2nd floor. For each day of the week beginning 2016-11-14 show how many rooms are being vacated that day by floor number. Show all days in the correct order.**

**SQL:**

SELECT DATE\_ADD(booking\_date,INTERVAL nights DAY) as i,SUM(left(room\_no,1)=1) AS 1st,SUM(left(room\_no,1)=2) AS 2nd,SUM(left(room\_no,1)=3) AS 3rd

FROM booking

WHERE DATE\_ADD(booking\_date,INTERVAL nights DAY) BETWEEN '2016-11-14' AND ' 2016-11-20'

GROUP BY DATE\_ADD(booking\_date,INTERVAL nights DAY)

Ejercicios en los que presentamos dificultad: 13, 14 y 15 de la sección *Hard questions*

### PARTE TRES. Definiendo e implementando consultas gerenciales

# Considerando la misión propuesta en este laboratorio (si lo requieren redefínanla) , definan e implementen la consulta más adecuada para que la organización conozca que tan bien está cumpliendo su misión. Justifíquenla como la mejor consulta.

COMO Administrador del hotel QUIERO conocer la concurrencia, en reservas, de un mismo cliente PARA PODER predecir su próxima reserva.

**Consulta en SQL:**

SELECT DISTINCT(guest\_id),COUNT(booking\_id)

FROM booking

GROUP BY guest\_id

# Proponga una pregunta, orientada a validar el logro en el cumplimiento de la misión, que no se pueda contestar actualmente. ¿Que cambios se deberían incluir en el modelo para poder responderla?

¿El potencial huésped canceló la reserva?

Debería incluir una tabla de estado de reservación (desde que reserva hasta que entrega el cuarto) con registros de tipo: completado, en proceso, cancelado.

# Considerando los tres usuarios detectados anteriormente, defina e implemente una consulta que le den información útil para cumplir con sus responsabilidades o satisfacer su necesidad.

# **Administrador del Hotel**

*COMO* Administrador del Hotel *QUIERO* conocer el flujo de clientes diario *PARA PODER* cubrir la atención en las reservas de forma adecuada.

**Situación ejemplo:** Flujo de clientes en la fecha: '2016-11-03' comparado con '2016-11-05'

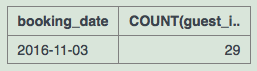
* + **SQL:**

SELECT booking\_date,COUNT(guest\_id)

FROM booking

WHERE booking\_date='2016-11-03'

GROUP BY booking\_date

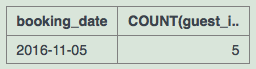


SELECT booking\_date,COUNT(guest\_id)

FROM booking

WHERE booking\_date='2016-11-05'

GROUP BY booking\_date



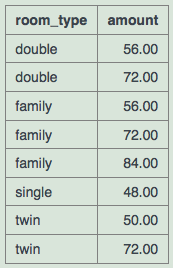
* **Plataforma de planes turísticos**

*COMO* Plataforma de planes turísticos *QUIERO* conocer los tipos de cuartos y tarifas *PARA PODER* ofrecer los cuartos del Hotel en la plataforma, correctamente.

* **SQL:**

SELECT room\_type,amount

FROM rate



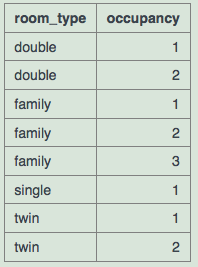
* **Huésped**

*COMO* Huésped *QUIERO* conocer la capacidad por tipo de cuarto *PARA PODER* elegir el cuarto que necesito.

* **SQL:**

SELECT room\_type,occupancy

FROM rate



Adicionalmente, ver archivo anexo en **guest.asta → (UseCase Diagram Part\_3)**

# RETROSPECTIVA

1. **¿Cuál fue el tiempo total invertido en el laboratorio por cada uno de ustedes? (Horas/Hombre)**

Aprox 10 ½ hrs / Laura Izquierdo

Aprox 8 hrs / Yarit Villalobos

1. **¿Cuál es el estado actual del laboratorio? ¿Por qué?**

Completo.

Nota: sin solución del punto 13, 14 y 15 de la parte dos, sección Hard questions.

Presentamos dificultad a la hora implementar por desconocimiento de otras Funciones u operadores.

1. **¿Cuál consideran fue el mayor logro? ¿Por qué?**

La adquisición de nuevas formas de realizar una consulta y el afianzamiento a la hora de implementarlo.

1. **¿Cuál consideran que fue el mayor problema técnico? ¿Qué hicieron para resolverlo?**

Entender cómo funcionaba Astah.

Para resolverlo, consultamos en internet documentación de SQL y vimos tutoriales.

1. **¿Qué hicieron bien como equipo? ¿Qué se comprometen a hacer para mejorar los resultados?**

Trabajar en conjunto y compartir formas de realizar consultas, discutir nuestras opiniones hasta llegar a un punto medio y así completar la implementación de las mismas.

Nos comprometemos a conocer más funciones y operadores, además de practicar, haciendo consultas; para así lograr consultas más eficientes.