

Universidad Nacional Autónoma de México

FACULTAD DE CIENCIAS

Equipo (=

HOTELES LUNA



Macal Cruz Brandon Brayan 318085470

Mayo 2024

## Introducción

En el contexto de la era digitalizada, la cadena de Hoteles Luna se enfrenta al desafío de modernizar su gestión de información. Actualmente, el uso de registros físicos y archivos no es óptimo para las necesidades actuales. Consciente de esta situación, el director ejecutivo de Hoteles Luna ha contratado a los alumnos de la Facultad de Ciencias para encontrar una solución que permita administrar los datos de manera sencilla, centralizada y consistente.

Como diseñadores de la Base de Datos (FBD), nuestra tarea es abordar todos los aspectos necesarios para garantizar que, en un futuro, la aplicación sea funcional a través de una plataforma web (etapa que está más allá del alcance de este proyecto). La confianza depositada por la cadena de hoteles en el diseño que proporcionaremos nos motiva a ofrecer un servicio mejorado a los clientes. Computólogos A.C., a quienes se nos ha adjudicado el contrato, presentará su solución el 03 de junio de 2024.

## Resumen de Objetivos

### Información del Hotel

Por lo que sabemos cada hotel debe almacenar la siguiente información:

- **Nombre del establecimiento:** El nombre único del hotel.
- **Dirección:** Incluye calle, número interior, número exterior, colonia y estado.
- **Teléfonos:** Los números de contacto del hotel.
- **Número total de habitaciones:** La cantidad total de habitaciones en el hotel.
- **Número de habitaciones disponibles:** Cuántas habitaciones están actualmente disponibles.
- **Número de habitaciones ocupadas:** Cuántas habitaciones están actualmente ocupadas.
- **Hora de check-in y check-out:** Los horarios para registrar la entrada y salida de los huéspedes.

### Servicios del Hotel

Cada hotel puede ofrecer diferentes servicios. Algunos ejemplos son piscina, estacionamiento, restaurante, bar, etc. Estos servicios deben registrarse de manera general para cada hotel.

### Tipos de Habitaciones

Se considerarán cuatro tipos de habitaciones:

- **Individuales:** Habitaciones para una sola persona.
- **Dobles:** Habitaciones con espacio para dos personas.
- **Cuádruples:** Habitaciones para cuatro personas.
- **Penthouse:** Habitaciones de lujo.

Cada tipo de habitación debe incluir:

- **Número de habitación:** Identificador único para cada habitación.
- **Número de camas:** Cuántas camas hay en la habitación.
- **Disponibilidad:** Si la habitación está disponible o no.
- **Costo por noche:** El precio de alojamiento por noche en esa habitación.

## Datos de los Huéspedes

Se deben almacenar los siguientes datos básicos de los huéspedes:

- **Nombre completo:** Incluye nombre, apellido paterno y apellido materno.
- **Nacionalidad:** País de origen del huésped.
- **Teléfonos:** Números de contacto del huésped.
- **Fecha de nacimiento:** La fecha en que nació el huésped.
- **Correo electrónico:** Dirección de correo electrónico del huésped.
- **Género:** Género del huésped (masculino, femenino, otro).

## Acciones propuestas

El equipo propuso un esquema en el que la base de datos podría trabajar de manera óptima donde se incluirían distintos campos para poder abarcar todos los datos del hotel

## Tablas

### 1. Hotel

- **idHotel (Llave primaria) | int**  
Un número entero que sirve como identificador único para cada hotel en la base de datos.
- **nombreEstablecimiento | varchar[(50)]**
- **Calle | varchar[(50)]**
- **NumInterior | int**
- **NumExterior | int**
- **Colonia | varchar[(50)]**
- **Estado | varchar[(50)]**
- **horaCheckIn | date**
- **horaCheckOut | date**
- **Servicio | varchar[(50)]**
- **PetFriendly | bool**

### 2. TelefonoHotel

- **idHotel | int**

- idTelefono(Llave primaria) | int(8)

### 3. CorreoHotel

- idHotel | int  
Llave de referencia al hotel
- CorreoHotel(Llave primaria) | int[(8)]

### 4. Huesped

- Atributos
  - idHuesped(Llave primaria) | int
  - nombre | varchar[(n)]
  - apellido paterno | varchar[(n)]
  - apellido materno | varchar[(n)]
  - Nacionalidad | varchar[(n)]
  - FechaNacimiento | date
  - Genero | varchar[(n)]

### 5. TelefonoHuesped

- Atributos
  - idTelefono(Llave primaria) | int

### 6. CorreoHuesped

- Atributos
  - CorreoCliente(Llave primaria) | varchar[(n)]

### 7. Habitacion (Doble, *Pent\_house*, Cuadruple, individual)

- Atributos
  - numHabitacion (Llave primaria) | int
  - idHotel (Llave foránea) | int
  - idEmpleado (Llave foránea) | int
  - NoCama | int
  - Disponible | bool
  - CostoPorNoche | int

### 8. Empleado (Apoyo, Recepcionista, Limpieza, Comida)

- Atributos
  - idEmpleado (Llave primaria) | int
  - RFC | varchar[(n)]
  - nombre | varchar[(n)]
  - apellido paterno | varchar[(n)]
  - apellido materno | varchar[(n)]
  - calle | varchar[(n)]
  - número interior | int
  - número exterior | int
  - colonia | varchar[(n)]

- estado | varchar[(n)]
- FechaContratacion | date
- FechaNacimiento | date
- Genero | varchar[(n)]
- Horario | date

#### 9. TelefonoEmpleado (Apoyo, Recepcionista, Limpieza, Comida)

- Atributos
  - idTelefono(Llave primaria) | int

#### 10. CorreoEmpleado (Apoyo, Recepcionista, Limpieza, Comida)

- Atributos
  - CorreoEmpleado(Llave primaria) | varchar[(n)]

#### 11. Salon

- Atributos
  - idSalon (Llave primaria) | int
  - idHuesped (Llave foránea) | int
  - Nombre | varchar[(n)]
  - calle | varchar[(n)]
  - número interior | int
  - número exterior | int
  - colonia | varchar[(n)]
  - estado | varchar[(n)]
  - Capacidad | int
  - HorarioAtencion | date
  - NombreSalon | varchar[(n)]
  - NoAsistente | varchar[(n)]
  - FechaEvento | date
  - NoMembresia | int
  - NombreHuesped | varchar[(n)]
  - HoraInicio | date
  - HoraFin | date

#### 12. Encuesta

- Atributos
  - idEncuesta (Llave primaria) | int
  - idHuesped (Llave foránea) | int
  - Instalacion | varchar[(n)]
  - PosibilidadRegreso | bool
  - ServicioDado | varchar[(n)]
  - Comentario | varchar[(n)]
  - Calificacion | int
    - a) Un número entre 0 y 10.
  - Recomendacion | varchar[(n)]
  - Instalacion | varchar[(n)]

#### 13. FormaPago

- Atributos

- IdPago (Llave primaria) | int
- NoTarjeta | int
- FechaVencimiento | varchar[(n)]
- Efectivo | int

## Consultas realizadas

Con la base de datos en operación, hemos podido realizar una serie de consultas que ilustran su uso y los datos que se pueden obtener. A continuación, presentamos una compilación de 15 consultas realizadas, junto con los datos que cada una de ellas puede proporcionar. Estas consultas destacan la capacidad de la base de datos para generar información valiosa y su potencial para satisfacer diversas necesidades de análisis y gestión.

### 1. - Cantidad de clientes hospedados por tipo de habitación

```
/*Consulta 1*/
SELECT TipoHabitacion, COUNT(idHuesped) AS CantidadClientes
FROM Recibo
WHERE
    Recibo.Checkout > '2024-06-03'
GROUP BY TipoHabitacion ;
```

En esta ocasión, a partir de la fecha actual, consultamos la cantidad de clientes por tipo de habitación para saber si aún siguen hospedados. Los resultados son los siguientes:

	tipohabitacion character varying (50) 🔒	cantidadclientes bigint 🔒
1	Individual	56
2	Cuadruple	49
3	Pent house	68
4	Doble	65

En ella, podemos ver una proporción equitativa en el número de clientes hospedados en distintos tipos de habitaciones.

### 2. - Cantidad de clientes hospedados por tipo de pago

```
/*Consulta 2*/
SELECT
    r.TipoPago,
    COUNT(r.idHuesped) AS CantidadClientes
FROM
    Recibo r
JOIN
    Huesped h ON r.idHuesped = h.idHuesped
WHERE
    r.Checkout > '2024-06-03'
GROUP BY
    r.TipoPago;
```

Esta consulta identifica cuántos clientes están hospedados según el tipo de pago utilizado. Los resultados son los siguientes:

	tipopago character varying (50)	cantidadclientes bigint
1	Efectivo	113
2	Tarjeta	125

Los datos muestran las preferencias de pago de los huéspedes.

### 3. - Cantidad de clientes hospedados por establecimiento

```
/*Consulta 3*/
SELECT
  h.NombreEstablecimiento,
  COUNT(r.idHuesped) AS CantidadClientes
FROM
  Hotel h
JOIN
  Recibo r ON h.idHotel = r.idHotel
WHERE
  r.Checkout > '2024-06-03'
GROUP BY
  h.NombreEstablecimiento;
```

En esta consulta se muestra la cantidad de clientes actualmente hospedados en cada establecimiento. Los resultados son los siguientes:

	nombrestablecimiento character varying (50)	cantidadclientes bigint
1	Village Green	1
2	Monica	5
3	Roth	4
4	Village	2
5	Welch	5
6	Mendota	3
7	Bowman	3
8	Northland	1
9	Schlimgen	3
10	Canary	4
11	Jana	5
12	Donald	3
13	Graceland	5
14	Kenwood	3

Podemos comparar la ocupación entre diferentes hoteles.

### 4. - Cantidad de clientes con membresía

```
/*Consulta 4*/
SELECT COUNT(*) AS CantidadClientesConMembresia
FROM Recibo
JOIN Huesped ON Recibo.idHuesped = Huesped.idHuesped
WHERE Recibo.esMiembro = true
```

Esta consulta determina la cantidad de clientes que tienen membresía. Los resultados son los siguientes:

	cantidadclientesconmembresia
1	185

Se identifica el alcance y popularidad de los programas de lealtad.

5. - Periodo de tiempo que han tenido su membresía

```
/*Consulta 5*/
SELECT
  R.idHuesped,
  MIN(R.CheckIn) AS FechaInicioMembresia,
  MAX(R.Checkout) AS FechaFinMembresia,
  (MAX(R.Checkout) - MIN(R.CheckIn)) AS PeriodoMembresiaDias
FROM Recibo R
INNER JOIN Huesped H ON R.idHuesped = H.idHuesped
WHERE R.esMiembro = true
GROUP BY R.idHuesped
order by idhuesped
```

Aquí se analiza el periodo de tiempo que los clientes han mantenido su membresía. Los resultados son los siguientes:

	idhuesped	fecha iniciomembresia	fecha finmembresia	periodoembresiadias
	integer	date	date	integer
1	1	2022-12-28	2024-11-21	694
2	2	2023-02-09	2024-08-18	556
3	4	2023-07-01	2024-09-17	444
4	5	2023-05-12	2024-12-09	577
5	6	2023-03-25	2024-10-23	578
6	7	2023-08-03	2024-06-18	320
7	8	2024-02-03	2024-11-30	301
8	10	2023-09-30	2024-08-05	310
9	11	2023-02-02	2024-02-10	373
10	12	2023-04-24	2024-11-14	570
11	14	2022-12-22	2024-11-22	701
12	15	2023-08-05	2024-08-24	385
13	16	2023-10-26	2024-10-17	357
14	17	2023-08-08	2024-09-24	413

Se obtiene información sobre la fidelidad y retención de clientes.

6. - Hoteles que son petfriendly



```
/*Consulta 6*/
SELECT NombreEstablecimiento, Calle, NumeroInterior, NumeroExterior, Colonia, Estado
FROM Hotel
WHERE PetFriendly = true;
```

Esta consulta lista los hoteles que permiten mascotas. Los resultados son los siguientes:

	nombreestablecimiento character varying (50)	calle character varying (50)	numerointerior integer	numeroexterior integer	colonia character varying (50)	estado character varying (50)
1	Rieder	Meadow Vale	5	94	Novick	TX
2	Kenwood	Truax	53	4	Miller	CO
3	Miller	Heffernan	12	56	Havey	MS
4	Straubel	Clove	40	8	Melrose	CA
5	Village	3rd	39	26	Monument	TX
6	Dahle	Melody	17	12	Blue Bill Park	OR
7	Westridge	Victoria	30	10	Reinke	DC
8	Atwood	Glendale	32	53	Fair Oaks	OH
9	Ilene	Sheridan	64	29	Knutson	KY
10	Wayridge	Fremont	14	26	Continental	IA
11	Forest Dale	Harper	60	52	Browning	CA
12	Utah	Rigney	19	17	Park Meadow	TX
13	Saint Paul	Oak	15	36	Troy	TX
14	Oneill	Texas	50	64	Clove	TN

Ayuda a los clientes que viajan con sus animales a encontrar alojamiento adecuado.

#### 7. - Cantidad de clientes con mascota hospedados

```
/*Consulta 7*/
SELECT COUNT(DISTINCT r.idHuesped) AS Cantidad_Clientes_Con_Mascota
FROM Recibo r
JOIN Hotel h ON r.idHotel = h.idHotel
WHERE r.Mascota = true
AND h.PetFriendly = true
AND r.Checkout <= '2024-06-03';
```

Se determina la cantidad de clientes actualmente hospedados que tienen mascotas. Los resultados son los siguientes:

	cantidad_clientes_con_mascota bigint
1	34

Ofrece una visión del uso de las instalaciones petfriendly.

#### 8. - Persona del apoyo que tiene a su cargo más huéspedes por tipo de habitación

```

/*Consulta 8*/
SELECT
    A.idEmpleado,
    A.Nombre,
    RH.TipoHabitacion,
    COUNT(*) AS NumHuespedes
FROM
    Apoyo A
JOIN
    Recibo R ON A.idHotel = R.idHotel
JOIN
    Huesped H ON R.idHuesped = H.idHuesped
JOIN
    (
        SELECT DISTINCT
            idHotel,
            TipoHabitacion
        FROM
            Recibo
    ) RH ON R.idHotel = RH.idHotel
GROUP BY
    A.idEmpleado,
    A.Nombre,
    RH.TipoHabitacion
ORDER BY
    NumHuespedes DESC,
    A.idEmpleado ASC;

```

Esta consulta identifica al personal de apoyo que tiene a su cargo el mayor número de huéspedes por tipo de habitación. Los resultados son los siguientes:

	idempleado integer	nombre character varying (50)	tipohabitacion character varying (50)	numhuespedes bigint
1	67	Hannah	Pent house	9
2	67	Hannah	Cuadruple	9
3	67	Hannah	Individual	9
4	67	Hannah	Doble	9
5	8	Rodrick	Pent house	8
6	8	Rodrick	Individual	8
7	8	Rodrick	Cuadruple	8
8	8	Rodrick	Doble	8
9	46	Gonzales	Individual	8
10	46	Gonzales	Pent house	8
11	46	Gonzales	Cuadruple	8
12	46	Gonzales	Doble	8
13	66	El	Doble	8
14	66	El	Cuadruple	8

Ayuda en la asignación de recursos y gestión del personal.

9. - Información de los clientes con relación visitas por año

```

/*Consulta 9*/
SELECT
  H.idHuesped AS idHuesped,
  H.Nombre AS NombreHuesped,
  H.ApellidoP AS ApellidoPaterno,
  H.ApellidoM AS ApellidoMaterno,
  EXTRACT(YEAR FROM R.CheckIn) AS Año,
  COUNT(R.idRecibo) AS VisitasPorAño
FROM
  Huesped H
LEFT JOIN
  Recibo R ON H.idHuesped = R.idHuesped
GROUP BY
  H.idHuesped, H.Nombre, H.ApellidoP, H.ApellidoM, EXTRACT(YEAR FROM R.CheckIn)
ORDER BY
  idHuesped ASC, Año, VisitasPorAño;

```

En esta consulta se analiza la frecuencia de las visitas de los clientes por año. Los resultados son los siguientes:

	idhuesped [PK] integer	nombrehuesped character varying (50)	apellidopaterno character varying (50)	apellidomaterno character varying (50)	año numeric	visitasporaño bigint
1	1	Ayn	Maude	Polo	2022	1
2	1	Ayn	Maude	Polo	2023	2
3	2	Verile	Lindeberg	Holdey	2023	5
4	3	Betteann	Geharke	Rowly	2023	2
5	3	Betteann	Geharke	Rowly	2024	1
6	4	Juline	Oland	Pardue	2023	2
7	4	Juline	Oland	Pardue	2024	3
8	5	Verla	Courtcliff	McKeighen	2023	3
9	5	Verla	Courtcliff	McKeighen	2024	1
10	6	Elbertine	Sommers	Argont	2023	2
11	7	Corby	Gask	Mackin	2023	5
12	7	Corby	Gask	Mackin	2024	1
13	8	Evan	Brandon	Brimmacombe	2023	1
14	8	Evan	Brandon	Brimmacombe	2024	1

Proporciona datos sobre la recurrencia anual de los huéspedes.

10. - Información de los clientes por visitas por trimestre

```

/*Consulta 10*/
SELECT
  H.idHuesped AS idHuesped,
  H.Nombre AS NombreHuesped,
  H.ApellidoP AS ApellidoPaterno,
  H.ApellidoM AS ApellidoMaterno,
  EXTRACT(YEAR FROM R.CheckIn) AS Año,
  CASE
    WHEN EXTRACT(MONTH FROM R.CheckIn) BETWEEN 1 AND 3 THEN 'Trimestre 1'
    WHEN EXTRACT(MONTH FROM R.CheckIn) BETWEEN 4 AND 6 THEN 'Trimestre 2'
    WHEN EXTRACT(MONTH FROM R.CheckIn) BETWEEN 7 AND 9 THEN 'Trimestre 3'
    WHEN EXTRACT(MONTH FROM R.CheckIn) BETWEEN 10 AND 12 THEN 'Trimestre 4'
  END AS Trimestre,
  COUNT(R.idRecibo) AS VisitasPorTrimestre
FROM
  Huesped H
LEFT JOIN
  Recibo R ON H.idHuesped = R.idHuesped
GROUP BY
  H.idHuesped, H.Nombre, H.ApellidoP, H.ApellidoM, EXTRACT(YEAR FROM R.CheckIn), Trimestre
ORDER BY
  idHuesped ASC, Año, Trimestre, VisitasPorTrimestre;

```

Aquí se muestra la cantidad de visitas de los clientes divididas por trimestre. Los resultados son los siguientes:

	idhuesped [PK] integer	nombrehuesped character varying (50)	apellidopaterno character varying (50)	apellidomaterno character varying (50)	año numeric	trimestre text	visitasportrimestre bigint
1	1	Ayn	Maude	Polo	2022	Trimestre 4	1
2	1	Ayn	Maude	Polo	2023	Trimestre 3	1
3	1	Ayn	Maude	Polo	2023	Trimestre 4	1
4	2	Verile	Lindeberg	Holdey	2023	Trimestre 1	2
5	2	Verile	Lindeberg	Holdey	2023	Trimestre 3	2
6	2	Verile	Lindeberg	Holdey	2023	Trimestre 4	1
7	3	Betteann	Geharke	Rowly	2023	Trimestre 2	1
8	3	Betteann	Geharke	Rowly	2023	Trimestre 4	1
9	3	Betteann	Geharke	Rowly	2024	Trimestre 1	1
10	4	Juline	Oland	Pardue	2023	Trimestre 1	1
11	4	Juline	Oland	Pardue	2023	Trimestre 3	1
12	4	Juline	Oland	Pardue	2024	Trimestre 1	3
13	5	Verla	Courtcliff	McKeighen	2023	Trimestre 2	2
14	5	Verla	Courtcliff	McKeighen	2023	Trimestre 3	1

Ayuda a identificar patrones estacionales y trimestrales.

11. - Información de los clientes por visitas por mes

```

/*Consulta 11*/
SELECT
  H.idHuesped AS idHuesped,
  H.Nombre AS NombreHuesped,
  H.ApellidoP AS ApellidoPaterno,
  H.ApellidoM AS ApellidoMaterno,
  EXTRACT(YEAR FROM R.CheckIn) AS Año,
  EXTRACT(MONTH FROM R.CheckIn) AS Mes,
  COUNT(R.idRecibo) AS VisitasPorMes
FROM
  Huesped H
LEFT JOIN
  Recibo R ON H.idHuesped = R.idHuesped
GROUP BY
  H.idHuesped, H.Nombre, H.ApellidoP, H.ApellidoM, EXTRACT(YEAR FROM R.CheckIn), EXTRACT(MONTH FROM R.CheckIn)
ORDER BY
  idHuesped ASC, Año, Mes, VisitasPorMes;

```

Se detalla la frecuencia de las visitas mensuales de los clientes. Los resultados son los siguientes:

	idhuesped [PK] integer	nombrehuesped character varying (50)	apellidopaterno character varying (50)	apellidomaterno character varying (50)	año numeric	mes numeric	visitaspormes bigint
1	1	Ayn	Maude	Polo	2022	12	1
2	1	Ayn	Maude	Polo	2023	8	1
3	1	Ayn	Maude	Polo	2023	11	1
4	2	Verile	Lindeberg	Holdey	2023	2	1
5	2	Verile	Lindeberg	Holdey	2023	3	1
6	2	Verile	Lindeberg	Holdey	2023	7	1
7	2	Verile	Lindeberg	Holdey	2023	9	1
8	2	Verile	Lindeberg	Holdey	2023	12	1
9	3	Betteann	Geharke	Rowly	2023	5	1
10	3	Betteann	Geharke	Rowly	2023	12	1
11	3	Betteann	Geharke	Rowly	2024	3	1
12	4	Juline	Oland	Pardue	2023	1	1
13	4	Juline	Oland	Pardue	2023	7	1
14	4	Juline	Oland	Pardue	2024	1	1

Permite un análisis más granular de la recurrencia.

## 12. - Información de los clientes por visitas por género

```

/*Consulta 12*/
SELECT
  H.Genero,
  COUNT(R.idRecibo) AS VisitasPorGenero
FROM
  Huesped H
LEFT JOIN
  Recibo R ON H.idHuesped = R.idHuesped
GROUP BY
  H.Genero
ORDER BY
  VisitasPorGenero DESC;

```

Esta consulta categoriza las visitas de los clientes por género. Los resultados son los siguientes:

	genero character varying (50)	visitasporgenero bigint
1	Male	183
2	Female	175
3	Non-binary	16
4	Genderfluid	10
5	Agender	9
6	Bigender	4
7	Genderqueer	3

Proporciona datos demográficos importantes para el análisis de mercado.

13. - Número de empleados de cada tipo por hotel

```

/*Consulta 13 : */
SELECT
    H.idHotel,
    H.NombreEstablecimiento,
    COUNT(A.idEmpleado) AS Apoyo,
    COUNT(R.idEmpleado) AS Recepcionista,
    COUNT(L.idEmpleado) AS Limpieza,
    COUNT(C.idEmpleado) AS Comida
FROM
    Hotel H
LEFT JOIN
    Apoyo A ON H.idHotel = A.idHotel
LEFT JOIN
    Recepcionista R ON H.idHotel = R.idHotel
LEFT JOIN
    Limpieza L ON H.idHotel = L.idHotel
LEFT JOIN
    Comida C ON H.idHotel = C.idHotel
GROUP BY
    H.idHotel, H.NombreEstablecimiento
ORDER BY
    H.idHotel;

```

En esta consulta se determina la cantidad de empleados por tipo (apoyo, limpieza, comida, recepcionista) en cada hotel. Los resultados son los siguientes:

	idhotel [PK] integer 	nombrestablecimiento character varying (50) 	apoyo bigint 	repcionista bigint 	limpieza bigint 	comida bigint 
1	1	Tony	5	0	0	0
2	2	Rieder	1	0	0	0
3	3	Kenwood	4	4	4	0
4	4	Swallow	3	0	3	0
5	5	Miller	1	1	0	0
6	6	Anzinger	0	1	0	1
7	7	Scofield	16	16	16	16
8	8	Butternut	16	16	16	0
9	9	Ludington	0	3	3	0
10	10	Straubel	2	0	0	2
11	11	Sunnyside	4	4	4	4
12	12	Village	0	1	0	1
13	13	Magdeline	4	4	4	0
14	14	Dahle	1	0	1	1

Ayuda en la gestión de recursos humanos.

14. - Edad promedio de los empleados de cada tipo (apoyo, limpieza, comida, recepcionista) por hotel

```
/*Consulta 14*/
SELECT
    H.idHotel,
    H.NombreEstablecimiento,
    'Apoyo' AS TipoEmpleado,
    AVG(EXTRACT(YEAR FROM AGE(CURRENT_DATE, A.FechaNacimiento))) AS PromedioEdad
FROM
    Hotel H
LEFT JOIN
    Apoyo A ON H.idHotel = A.idHotel
GROUP BY
    H.idHotel, H.NombreEstablecimiento

UNION ALL

SELECT
    H.idHotel,
    H.NombreEstablecimiento,
    'Recepcionista' AS TipoEmpleado,
    AVG(EXTRACT(YEAR FROM AGE(CURRENT_DATE, R.FechaNacimiento))) AS PromedioEdad
FROM
    Hotel H
LEFT JOIN
    Recepcionista R ON H.idHotel = R.idHotel
GROUP BY
    H.idHotel, H.NombreEstablecimiento

UNION ALL

SELECT
    H.idHotel,
    H.NombreEstablecimiento,
    'Limpieza' AS TipoEmpleado,
    AVG(EXTRACT(YEAR FROM AGE(CURRENT_DATE, L.FechaNacimiento))) AS PromedioEdad
FROM
    Hotel H
LEFT JOIN
    Limpieza L ON H.idHotel = L.idHotel
GROUP BY
    H.idHotel, H.NombreEstablecimiento

UNION ALL

SELECT
    H.idHotel,
    H.NombreEstablecimiento,
    'Comida' AS TipoEmpleado,
    AVG(EXTRACT(YEAR FROM AGE(CURRENT_DATE, C.FechaNacimiento))) AS PromedioEdad
FROM
    Hotel H
LEFT JOIN
    Comida C ON H.idHotel = C.idHotel
GROUP BY
    H.idHotel, H.NombreEstablecimiento
ORDER BY
    idHotel, TipoEmpleado;
```

Aquí se analiza la edad promedio de los empleados según su rol en cada hotel. Los resultados son los siguientes:



	idhotel integer	nombrestablecimiento character varying (50)	tipoempleado text	promedioedad numeric
1	1	Tony	Apoyo	56.2000000000000000
2	1	Tony	Comida	[null]
3	1	Tony	Limpieza	[null]
4	1	Tony	Recepcionista	[null]
5	2	Rieder	Apoyo	38.0000000000000000
6	2	Rieder	Comida	[null]
7	2	Rieder	Limpieza	[null]
8	2	Rieder	Recepcionista	[null]
9	3	Kenwood	Apoyo	70.0000000000000000
10	3	Kenwood	Comida	[null]
11	3	Kenwood	Limpieza	50.7500000000000000
12	3	Kenwood	Recepcionista	56.0000000000000000
13	4	Swallow	Apoyo	43.0000000000000000
14	4	Swallow	Comida	[null]

Ayuda en la planificación y políticas de recursos humanos.

15. - Género de los empleados de cada tipo (apoyo, limpieza, comida, recepcionista) por hotel  
Esta consulta clasifica a los empleados por género y tipo de rol en cada hotel. Los resultados son los siguientes:

	idhotel integer	nombrestablecimiento character varying (50)	tipoempleado text	genero character varying (50)	cantidadempleados bigint
1	1	Tony	Apoyo	Agender	1
2	1	Tony	Apoyo	Female	2
3	1	Tony	Apoyo	Male	2
4	1	Tony	Comida	[null]	0
5	1	Tony	Limpieza	[null]	0
6	1	Tony	Recepcionista	[null]	0
7	2	Rieder	Apoyo	Female	1
8	2	Rieder	Comida	[null]	0
9	2	Rieder	Limpieza	[null]	0
10	2	Rieder	Recepcionista	[null]	0
11	3	Kenwood	Apoyo	Female	1
12	3	Kenwood	Comida	[null]	0
13	3	Kenwood	Limpieza	Female	1
14	3	Kenwood	Limpieza	Male	2

Proporciona datos demográficos para la gestión del personal.

16. - Edad promedio y nacionalidades de cada cliente en diferente hotel

```

/*Consulta 16*/
SELECT
  H.idHotel,
  H.NombreEstablecimiento,
  AVG(EXTRACT(YEAR FROM AGE(Hues.FechaNacimiento))) AS EdadPromedio,
  COUNT(*) AS TotalHuespedes,
  COUNT(DISTINCT Hues.Nacionalidad) AS TotalNacionalidades,
  STRING_AGG(DISTINCT Hues.Nacionalidad, ', ') AS Nacionalidades
FROM
  Hotel H
JOIN
  Recibo R ON H.idHotel = R.idHotel
JOIN
  Huesped Hues ON R.idHuesped = Hues.idHuesped
GROUP BY
  H.idHotel, H.NombreEstablecimiento
ORDER BY
  H.idHotel;

```

Finalmente, se analiza la edad promedio y las nacionalidades de los clientes en diferentes hoteles. Los resultados son los siguientes:

	idhotel [PK] integer	nombrestablecimiento character varying (50)	edadpromedio numeric	totalhuespedes bigint	totalnacionalidades bigint	nacionalidades text
1	1	Tony	13.5000000000000000	4	3	Argentina, China, Russia
2	2	Rieder	13.0000000000000000	3	3	Albania, China, Philippines
3	3	Kenwood	20.4000000000000000	5	5	Germany, Indonesia, Philippines, Poland, Portugal
4	4	Swallow	24.5000000000000000	2	2	Sweden, Ukraine
5	5	Miller	12.3333333333333333	3	3	China, Poland, Zambia
6	6	Anzinger	10.6666666666666667	3	3	Brazil, Portugal, Zambia
7	7	Scofield	19.0000000000000000	4	3	Indonesia, Poland, Russia
8	8	Butternut	18.5000000000000000	2	2	Philippines, Ukraine
9	9	Ludington	22.0000000000000000	4	4	China, Russia, South Korea, Ukraine
10	10	Straubel	15.0000000000000000	3	3	China, Poland, Ukraine
11	11	Sunnyside	20.3333333333333333	3	3	China, Indonesia, Philippines
12	12	Village	22.0000000000000000	3	3	China, Russia, Sweden
13	13	Magdeline	19.5000000000000000	2	2	Peru, Serbia
14	14	Dahle	16.0000000000000000	3	2	Japan, Thailand

Proporciona una visión demográfica detallada de los huéspedes.