



**Proyecto: Parque de Diversiones**

**Integrantes:**

**Mauricio Diaz Villarreal- 200854**

**Gerardo Armando Guerrero Alvarez - 203214**

**Santiago Villaseñor Ramírez- 201427**

**Primavera 2023**

**Materia: Bases de Datos**

**Profesor: Felipe Lopez**

<b>Primera Parte</b>	<b>3</b>
<b>Definición</b>	<b>3</b>
Entidades	3
<b>Diagrama Entidad-Vínculo</b>	<b>4</b>
<b>Creación de tablas en SQL</b>	<b>5</b>
<b>Inserción de tuplas en SQL</b>	<b>7</b>
<b>Consultas en SQL</b>	<b>11</b>
<b>Segunda Parte</b>	<b>13</b>
– (Atracciones anuales) Obtener el nombre de los visitantes que subieron a atracciones y contar cuántas veces subieron en 2022 y lo que va de 2023:	13
– Visitas históricas a cada área	14
– Monto de las ventas por año	15
– Porcentaje de cada tipo de boletos (Considerando el total):	15
Consultas realizadas en Python	16
Especificaciones del hardware y software utilizados:	18
Conclusiones	20

## Primera Parte

### Definición

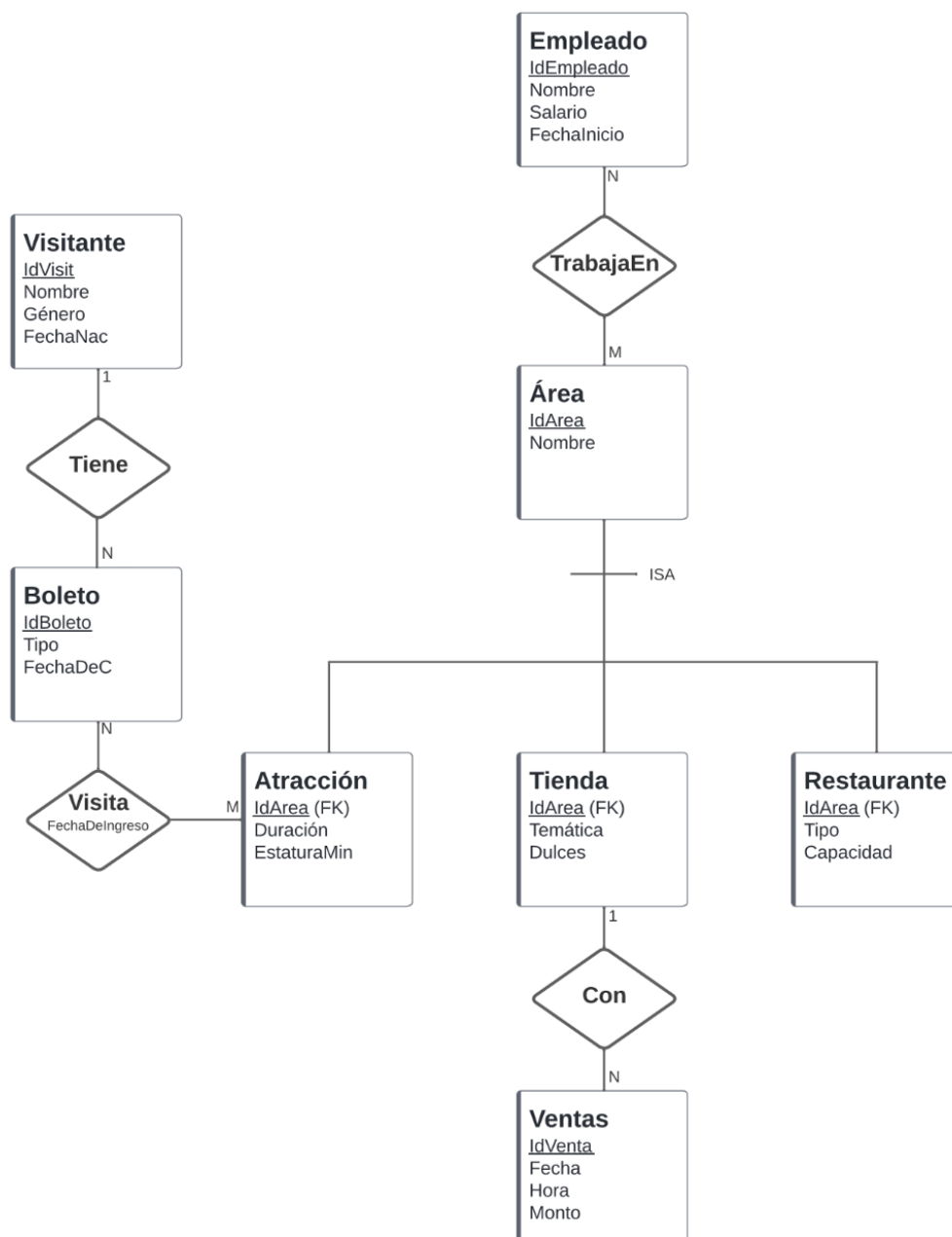
*Se hará una base de datos para un parque de diversiones. Dicho parque tendrá tres áreas distintas: atracciones, tiendas y restaurantes. Los empleados que contratan pueden trabajar en una o varias de estas áreas.* El parque tendrá un nombre, dirección, capacidad y una clave que lo identifique. El parque registra los visitantes, que entran con un boleto. Por cada visitante se obtiene su nombre, rango de edad y género. Un boleto puede ser de varios tipos, que brindan distintos beneficios, y una fecha de compra. Con el boleto, el cliente

tendrá acceso al parque y se registrará la fecha de entrada. Los empleados cuentan con una clave de identificación, nombre, salario, fecha de inicio y en las áreas en que trabaja. El área de trabajo tendrá un nombre y puede ser una atracción, tienda o restaurante. La atracción tendrá su propia clave, la duración del juego y la estatura mínima para poder usar las instalaciones. Los restaurantes tendrán una clave que lo identifique, el tipo de comida que servirá y la capacidad total. La tienda tendrá una clave, la temática y si tiene venta de dulces. Esta tendrá ventas que se identificarán con una clave, tendrán fecha de realización, el monto y la hora en la que se hizo la compra.

## Entidades

- Parque
- Boleto
- Visitante
- Empleado
- Área
- Restaurante
- Atracción
- Tienda
- Ventas

## Diagrama Entidad-Vínculo



## Creación de tablas en SQL

--visitante

```
create table Visitante
(
    IdVisit int primary key,
    Nombre varchar(30),
    Género varchar(4) check(Género in ('hom', 'muj', 'otro')),
    FechaNac date
);
```

--boleto

```
create table Boleto
(
    IdBoleto int primary key,
    Tipo varchar(30) check(Tipo in ('Básico', 'Premium')),
    FechaDeC date,
    IdVisit int references Visitante unique
);
```

--área

```
create table Área
(
    IdArea smallint primary key,
    Nombre varchar(30)
);
```

--atracción

```
create table Atracción
(
    IdArea smallint references Área primary key,
    Duración int,
    EstaturaMin real
);
```

--visita

```
create table Visita
(
    IdBoleto int references Boleto,
    IdArea smallint references Área,
    primary key(IdBoleto, IdArea),
    FechaDeIngreso date
);
```

--empleado

```
create table Empleado
(
    IdEmpleado smallint primary key,
    Nombre varchar(30),
    Salario money,
    FechaInicio date
);
```

--trabajaEn

```
create table TrabajaEn
(
    IdEmpleado smallint references Empleado,
    IdArea smallint references Área,
                                primary key(IdEmpleado, IdArea)
);
```

--tienda

```
create table Tienda
(
    IdArea smallint references Área primary key,
    Temática varchar(30),
    Dulces boolean
);
```

--restaurante

```
create table Restaurante
(
    IdArea smallint references Área primary key,
    Tipo varchar(30) check(Tipo in ('USA', 'MEX', 'JAP')),
    Capacidad smallint
);
```

--ventas

```
create table Venta
(
    IdVenta int primary key,
    Fecha date,
    Hora time,
    Monto real,
    IdArea smallint references Área
)
```

## Inserción de tuplas en SQL

--tuplas del visitante

```
insert into Visitante values (10100,'Juan','hom','2000-12-10');
insert into Visitante values (10101,'Isabel','muj','1990-01-20');
insert into Visitante values (10102,'Mauricio','otro','1940-07-14');
insert into Visitante values (10103,'Debbie','muj','1997-06-21');
insert into Visitante values (10104,'Jason','hom','2004-06-27');
insert into Visitante values (10105,'Santiago','hom','2002-05-09');
```

–NUEVAS:

```
insert into Visitante values (10106, 'María', 'muj', '1995-09-18');
insert into Visitante values (10107, 'Carlos', 'hom', '1988-03-12');
insert into Visitante values (10108, Gerardo, 'hom', '2000-05-04');
insert into Visitante values (10109, 'Pedro', 'hom', '1979-07-05');
insert into Visitante values (10110, 'Laura', 'muj', '1985-02-09');
```

--tuplas de boletos

```
insert into Boleto values (30310, 'Básico', '2022-02-25', 10100);
insert into Boleto values (40121, 'Premium', '2021-03-19', 10101);
insert into Boleto values (25312, 'Premium', '2020-03-16', 10102);
insert into Boleto values (31313, 'Básico', '2022-11-29', 10103);
insert into Boleto values (45214, 'Básico', '2023-03-12', 10104);
insert into Boleto values (11215, 'Básico', '2022-12-28', 10105);
```

```
insert into Boleto values (45215, 'Básico', '2022-08-02', 10100);
insert into Boleto values (30312, 'Básico', '2023-05-30', 10102);
insert into Boleto values (40123, 'Básico', '2022-10-18', 10103);
insert into Boleto values (25314, 'Premium', '2023-03-08', 10104);
insert into Boleto values (58234, 'Premium', '2023-08-15', 10104);
insert into Boleto values (43285, 'Premium', '2023-02-14', 10105);
insert into Boleto values (31315, 'Básico', '2022-07-22', 10105);
insert into Boleto values (29746, 'Básico', '2022-08-15', 10106);
insert into Boleto values (69696, 'Premium', '2023-02-14', 10106);
insert into Boleto values (40122, 'Básico', '2022-09-15', 10107);
insert into Boleto values (11216, 'Premium', '2023-02-14', 10107);
insert into Boleto values (25313, 'Premium', '2023-04-20', 10108);
insert into Boleto values (29746, 'Básico', '2023-08-15', 10108);
insert into Boleto values (31314, 'Básico', '2023-06-10', 10109);
```



--tuplas para área

```
insert into Área values (097, 'SuperSonic');
insert into Área values (072, 'Montaña del terror');
insert into Área values (018, 'Velocity');
insert into Área values (130, 'Rudy s');
insert into Área values (154, 'Sushi Inn');
insert into Área values (187, 'Tacos de Toño');
insert into Área values (243, 'Regalos Zeus');
```

```
insert into Área values (013, 'El demonizador');
insert into Área values (287, 'Mundo de peluche');
```

--tuplas de Atracción

```
insert into Atracción values (097, 60, 1.20);
insert into Atracción values (072, 80, 1.60);
insert into Atracción values (018, 100, 1.60);
```

```
insert into Atracción values (013,112,1.65)
```

--tuplas para visitas

```
insert into Visita values (30310,097, '2021-01-24');
insert into Visita values (30310,072, '2021-01-24');
insert into Visita values (40121,018, '2022-11-04');
insert into Visita values (25312,072, '2020-07-27');
insert into Visita values (31313,097, '2021-05-09');
insert into Visita values (40121,072, '2022-11-04');
insert into Visita values (11215,018, '2022-11-12');
insert into Visita values (45214,097, '2021-08-25');
```

–Estas son visitas a atracciones

```
insert into Visita values (45215, 072, '2022-08-02');
insert into Visita values (30312, 013, '2023-05-30');
insert into Visita values (40123, 072, '2022-10-18');
insert into Visita values (25314, 097, '2023-03-08');
insert into Visita values (58234, 018, '2023-08-15');
insert into Visita values (43285, 072, '2023-02-14');
insert into Visita values (31315, 013, '2022-07-22');
insert into Visita values (29746, 072, '2022-08-15');
insert into Visita values (69696, 097, '2023-02-14');
insert into Visita values (40122, 018, '2022-09-15');
```

```
insert into Visita values (11216, 097, '2023-02-14');
insert into Visita values (25313, 018, '2023-04-20');
insert into Visita values (29746, 013, '2023-08-15');
insert into Visita values (31314, 072, '2023-06-10');
```

–Estas son tuplas de visitas a restaurantes

```
insert into Visita values (30310, 154, '2021-01-24');
insert into Visita values (30310, 187, '2021-01-24');
insert into Visita values (40121, 243, '2022-11-04');
insert into Visita values (25312, 187, '2020-07-27');
insert into Visita values (31313, 154, '2021-05-09');
insert into Visita values (40121, 187, '2022-11-04');
insert into Visita values (11215, 243, '2022-11-12');
insert into Visita values (45214, 154, '2021-08-25');
```

```
insert into Visita values (45215, 187, '2022-08-02');
insert into Visita values (30312, 243, '2023-05-30');
insert into Visita values (40123, 187, '2022-10-18');
insert into Visita values (25314, 154, '2023-03-08');
insert into Visita values (58234, 243, '2023-08-15');
insert into Visita values (43285, 187, '2023-02-14');
insert into Visita values (31315, 243, '2022-07-22');
insert into Visita values (29746, 187, '2022-08-15');
insert into Visita values (69696, 154, '2023-02-14');
insert into Visita values (40122, 243, '2022-09-15');
insert into Visita values (11216, 154, '2023-02-14');
insert into Visita values (25313, 243, '2023-04-20');
insert into Visita values (29746, 154, '2023-08-15');
insert into Visita values (31314, 187, '2023-06-10');
```

--tuplas para empleado

```
insert into Empleado values (1234, 'Constanza', 3000, '2019-01-05');
insert into Empleado values (1986, 'Diego', 21100, '2020-03-03');
insert into Empleado values (0988, 'Lucia', 6000, '2019-02-11');
insert into Empleado values (0876, 'Monica', 30000, '2018-05-22');
insert into Empleado values (1544, 'Elmer', 5000, '2022-08-30');
insert into Empleado values (0923, 'Luisa', 25000, '2019-11-14');
insert into Empleado values (2010, 'Sergio', 8000, '2021-03-10');
insert into Empleado values (3365, 'Ana', 45000, '2017-09-02');
```

--tuplas para trabajaEn

```
insert into TrabajaEn values (1234, 097);
insert into TrabajaEn values (1986, 018);
insert into TrabajaEn values (0988, 130);
insert into TrabajaEn values (0876, 243);
insert into TrabajaEn values (1234, 187);
insert into TrabajaEn values (1544, 187);
```

```
insert into TrabajaEn values (1986, 072);
insert into TrabajaEn values (0988, 018);
insert into TrabajaEn values (0876, 130);
insert into TrabajaEn values (1544, 154);
insert into TrabajaEn values (0923, 187);
insert into TrabajaEn values (2010, 243);
insert into TrabajaEn values (3365, 013);
```

--tuplas para tienda

```
insert into tienda values (243, 'Dioses Griegos', false);
```

```
insert into tienda values (287, 'Muñecos de peluche', true)
```

--tuplas para restaurante

```
insert into Restaurante values (130, 'USA', 70);
insert into Restaurante values (154, 'JAP', 60);
insert into Restaurante values (187, 'MEX', 100);
```

--tuplas para ventas

```
insert into Venta values (20499, '2021-01-13', '18:00', 100, 243);
insert into Venta values (22692, '2020-12-09', '12:00', 250, 243);
insert into Venta values (11264, '2021-10-19', '13:00', 123, 243);
insert into Venta values (31586, '2022-01-17', '14:00', 54, 243);
insert into Venta values (13256, '2020-09-25', '15:00', 63, 243);
insert into Venta values (17643, '2022-11-28', '16:00', 346, 243);
```

```
insert into Venta values (24710, '2021-01-24', '12:34', 1345, 243);
insert into Venta values (67891, '2021-01-24', '14:56', 1230, 287);
insert into Venta values (45287, '2022-11-04', '10:23', 1050, 287);
insert into Venta values (89123, '2020-07-27', '18:43', 1950, 243);
insert into Venta values (57643, '2021-05-09', '11:09', 1400, 287);
insert into Venta values (12345, '2022-11-04', '15:30', 1320, 287);
insert into Venta values (98765, '2022-11-12', '19:15', 980, 243);
insert into Venta values (34671, '2021-08-25', '16:50', 1375, 243);
insert into Venta values (23456, '2022-08-02', '13:22', 1260, 287);
insert into Venta values (91234, '2023-05-30', '20:05', 945, 243);
```

```

insert into Venta values (87521, '2022-10-18', '09:40', 1280, 287);
insert into Venta values (98712, '2023-03-08', '11:55', 1485, 287);
insert into Venta values (54123, '2023-08-15', '17:18', 910, 243);
insert into Venta values (12398, '2023-02-14', '13:35', 1020, 287);
insert into Venta values (65743, '2022-07-22', '16:40', 1450, 243);
insert into Venta values (76521, '2022-08-15', '21:00', 780, 287);
insert into Venta values (78901, '2023-02-14', '09:57', 930, 287);
insert into Venta values (43267, '2022-09-15', '14:26', 1160, 243);
insert into Venta values (76532, '2023-02-14', '18:12', 1380, 243);
insert into Venta values (45678, '2023-04-20', '10:38', 1355, 287);
insert into Venta values (34567, '2023-08-15', '15:52', 980, 243);
insert into Venta values (90876, '2023-06-10', '12:17', 1270, 287);

```

## Consultas en SQL

--a) Una que involucre varias tablas y order by.

--Lista a los visitantes y ordénalos por alfabéticamente

```

select distinct v.nombre
from visitante v, boleto b, atracción a, visita vi
where v.IdVisit=b.IdVisit and b.IdBoleto=vi.IdBoleto and vi.IdArea=a.IdArea
order by v.nombre

```

--b) Una con manejo de fechas.

--ID de boletos con fechaDeCompra de este año

```

select IdBoleto
from Boleto
where FechaDeC >= '2023-01-01'

```

--c) Una con not in y subconsulta.

--Visitantes que no hayan ido a la atracción "Velocity"

```

select distinct nombre, boleto.idboleto
from visitante,visita,boleto
where visitante.IdVisit=boleto.IdVisit and visita.IdBoleto = boleto.IdBoleto and nombre not in
(select v.nombre
from boleto b, visita, Área a,visitante v
where b.idVisit = v.idVisit and b.IdBoleto = visita.IdBoleto
and visita.IdArea = a.IdArea and a.nombre='Velocity')

```

--d)Una con intersección.

--Empleados que trabajan en una atracción y restaurante

```

select e.nombre
from empleado e, trabajaen t, restaurante r
where e.IdEmpleado=t.IdEmpleado and t.IdArea=r.IdArea
intersect

```

```
select e.nombre
from empleado e, trabajaen t, atracción at
where e.IdEmpleado=t.IdEmpleado and t.IdArea=at.IdArea
```

--e) Una con unión.

--Empleados que trabajan en una tienda o restaurante

```
select e.nombre
from empleado e, trabajaen t, restaurante r
where e.IdEmpleado=t.IdEmpleado and t.IdArea=r.IdArea
union
select e.nombre
from empleado e, trabajaen t, tienda ti
where e.IdEmpleado=t.IdEmpleado and t.IdArea=ti.IdArea
```

--f) Una con agrupamiento.

-- Cuenta a cuántas atracciones se ha subido cada visitante

```
select v.nombre, count(*) "Número de atracciones"
from visitante v, boleto b, atracción a, visita vi
where v.IdVisit=b.IdVisit and b.IdBoleto=vi.IdBoleto and vi.IdArea=a.IdArea
group by v.nombre
```

--g) Una con agrupamiento y having.

--Visitantes que han subido a al menos 2 atracciones

```
select nombre, count(*) "Numero de atracciones"
from visitante v, boleto b, atracción a, visita vi
where v.IdVisit=b.IdVisit and b.IdBoleto=vi.IdBoleto and vi.IdArea=a.IdArea
group by visitante
having count(*) >= 2
```

--h) Una que contenga where (con al menos una condición que no sea sólo equijunta), group by y having.

--Empleados en más de un área y que tengan ganen más de 3000 pesos al mes

```
select e.nombre
from Área a, empleado e, trabajaen
where e.idempleado = trabajaen.idempleado and trabajaen.idarea = a.idarea and
salario>2500
group by e.nombre
having count(*)>=2
```

--i) Una que contenga el máximo, o mínimo, de un conjunto de grupos

--Estatura mínima para subir a al menos una atracción

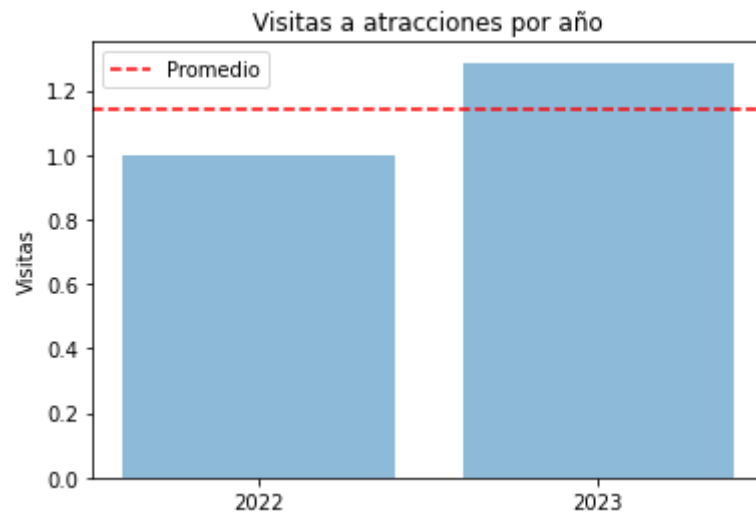
```
select min(estaturamin)
from atracción
```

## Segunda Parte

– (Atracciones anuales) Obtener el nombre de los visitantes que subieron a atracciones y contar cuántas veces subieron en 2022 y lo que va de 2023:

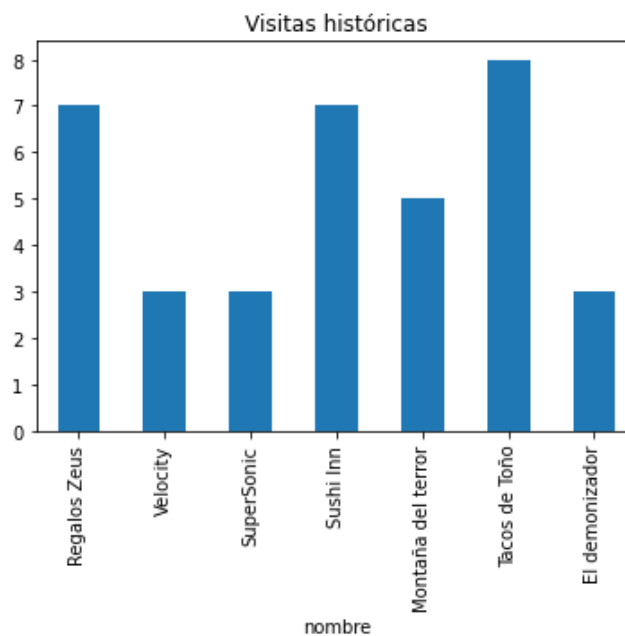
```
select nombre,count(*) as "visitas"
from visitante
join boleto
on boleto.idvisit = visitante.idvisit
where idboleto in(
    select idboleto
    from visita
    where idarea in (
        select idarea
        from atracción) and extract(year from fechadeingreso) = 2022)
group by nombre")
```

```
select nombre,count(*) as "visitas"
from visitante
join boleto
on boleto.idvisit = visitante.idvisit
where idboleto in(
    select idboleto
    from visita
    where idarea in (
        select idarea
        from atracción) and extract(year from fechadeingreso) = 2023)
group by nombre
```



– Visitas históricas a cada área

```
select nombre, count(*)
from visita b
join Área
on Área.idarea=b.idarea
group by nombre
```

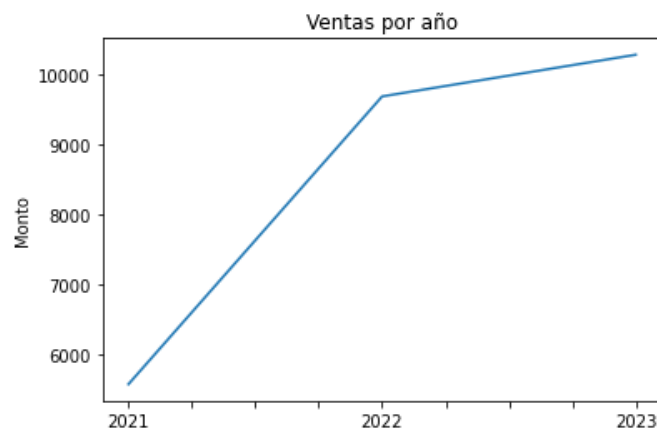


– Monto de las ventas por año

```
select sum(monto)
from venta
where extract(year from fecha) = 2023
```

```
select sum(monto)
from venta
where extract(year from fecha) = 2022
```

```
select sum(monto)
from venta
where extract(year from fecha) = 2021'
```

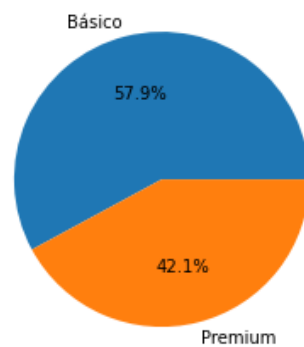


– Porcentaje de cada tipo de boletos (Considerando el total):

```
select count(*)
from boleto
where tipo = 'Básico'
```

```
select count(*)
from boleto
where tipo = 'Premium'
```

Marketshare de cada tipo de boleto





## Consultas realizadas en Python

```

import bibBD as lector
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import psycopg2 as obg

# Establecer la conexión a la base de datos
host = 'localhost'
database = 'Parque Diversiones 1.0'
user = 'postgres'
password = 'basesdatos_cd'

conex = lector.conexión(host, database, user, password)

# Consulta: visitas a atracciones en 2022 y 2023
df = lector.consult(conex, """
select nombre,count(*) as "visitas"
from visitante
join boleto
on boleto.idvisit = visitante.idvisit
where idboleto in(
    select idboleto
    from visita
    where idarea in (
        select idarea
        from atracción) and extract(year from fechadeingreso) = 2022)
group by nombre""")
atrac_22 = df['visitas']

df = lector.consult(conex, """
select nombre,count(*) as "visitas"
from visitante
join boleto
on boleto.idvisit = visitante.idvisit
where idboleto in(
    select idboleto
    from visita
    where idarea in (
        select idarea
        from atracción) and extract(year from fechadeingreso) = 2023)
group by nombre""")
atrac_23 = df['visitas']

# Datos de ejemplo para las barras
valores = [atrac_22.mean(), atrac_23.mean()]

```

```

# Valor de referencia
valor_referencia = np.mean(valores)

indices = np.arange(len(valores))
fig, ax = plt.subplots()

# Crea las barras
ax.bar(indices, valores, align='center', alpha=0.5)

# Crea la barra de referencia
ax.axhline(y=valor_referencia, color='r', linestyle='--', label='Promedio')

# Etiquetas en el eje x
ax.set_xticks(indices)
ax.set_xticklabels(['2022', '2023'])

# Etiquetas y título
ax.set_ylabel('Visitas')
ax.set_title('Visitas a atracciones por año')

# Leyendas
ax.legend()

plt.show()

# Visitas históricas de cada área
df = lector.consult(conex, """select nombre, count(*)
from visita b
join Área
on Área.idarea=b.idarea
group by nombre""")
df = df.set_index('nombre')
df = df['count']
print(df)
serie = df
ax = serie.plot.bar()
# Cambiar el nombre de la gráfica
ax.set_title('Visitas históricas')
# Mostrar el gráfico
plt.show()

# Evolución de las ventas anuales
df = lector.consult(conex, """select sum(monto)
from venta
where extract(year from fecha) = 2023""")
sum23 = df['sum']
sum23 = sum23[0]

```

```

print(sum23)
df = lector.consult(conex, "select sum(monto)
from venta
where extract(year from fecha) = 2022")
sum22 = df['sum']
sum22 = sum22[0]
print(sum22)
df = lector.consult(conex, "select sum(monto)
from venta
where extract(year from fecha) = 2021")
sum21 = df['sum']
sum21 = sum21[0]
print(sum21)

ventas = pd.Series({'2021':sum21,'2022':sum22,'2023':sum23})
ax = ventas.plot()
ax.set_ylabel('Monto')
ax.set_title('Ventas por año')


df = lector.consult(conex, "
select count(*)
from boleto
where tipo = 'Básico' ")
bas=df['count']
bas = bas[0]

df = lector.consult(conex, "
select count(*)
from boleto
where tipo = 'Premium' ")
premium=df['count']
premium = premium[0]

serie = pd.Series({'Básico':bas,'Premium':premium})
ax = serie.plot.pie(autopct='%1f%%')
ax.set_title('Marketshare de cada tipo de boleto')

plt.show()

```

## Especificaciones del hardware y software utilizados:

Procesador: Chip M1 de Apple con arquitectura ARM de 8 núcleos (CPU de 8 núcleos y GPU de 8 núcleos).

Memoria RAM: Puede venir con opciones de 8 GB o 16 GB de memoria unificada LPDDR4X.

Almacenamiento: Ofrece opciones de almacenamiento SSD de 256 GB, 512 GB, 1 TB o 2 TB.

Gráficos: GPU integrada con hasta 8 núcleos, capaz de ofrecer un rendimiento gráfico rápido y eficiente.

Batería: Batería interna de polímero de litio

Conectividad inalámbrica: Wi-Fi 6 (802.11ax) y Bluetooth 5.0.

Sistema operativo: Viene preinstalado con macOS Big Sur y es compatible con las últimas versiones de macOS.

## Conclusiones

El manejo de SQL a través de Python y el uso de Matplotlib son dos habilidades fundamentales para cualquier persona que trabaje con datos y análisis: científicos de datos. Estas herramientas permiten la manipulación de datos almacenados en bases de datos relacionales mediante consultas SQL y la visualización de esos datos de manera efectiva a través de gráficos y visualizaciones con Matplotlib.

La importancia del manejo de SQL a través de Python radica en que SQL es el lenguaje estándar para interactuar con bases de datos relacionales. Python, por su parte, es un lenguaje de programación versátil y popular que ofrece numerosas bibliotecas y módulos para conectarse a bases de datos y ejecutar consultas SQL. Al utilizar Python para manejar SQL, se pueden realizar tareas como la extracción, manipulación y transformación de datos de manera más eficiente y automatizada. Esto es especialmente útil cuando se trabaja con grandes volúmenes de datos o se necesitan realizar operaciones complejas en la base de datos.

Por otro lado, Matplotlib es una biblioteca de visualización en Python que permite crear una amplia gama de gráficos y visualizaciones. Esta herramienta es esencial para explorar y comunicar datos de manera efectiva, ya que proporciona opciones flexibles y personalizables para crear gráficos de líneas, barras, dispersión, histogramas, entre otros. Al combinar el manejo de SQL con Python y Matplotlib, se pueden obtener datos de la base de datos mediante consultas SQL y luego visualizarlos de manera clara y comprensible.

El uso de Matplotlib junto con SQL en Python permite a los analistas y científicos de datos extraer información valiosa de las bases de datos y comunicar los resultados de manera efectiva a través de gráficos. Esto facilita la identificación de patrones, tendencias y relaciones en los datos, lo que a su vez permite la toma de decisiones informadas y respaldadas por evidencia.

Al dominar estas habilidades, se puede realizar análisis de datos más sólidos y obtener una comprensión más profunda de la información contenida en las bases de datos.