



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

Facultad de Ingeniería



Materia: Base de Datos

GRUPO: 01

EQUIPO: 04

INTEGRANTES:

Castillo Soto Jacqueline
Diaz de Leon Jesus Alejandro
Figueroa Ruiz Carolina
Isidro Castro Karen Cristina

PROYECTO FINAL

fecha de entrega: 27 de Mayo de 2024

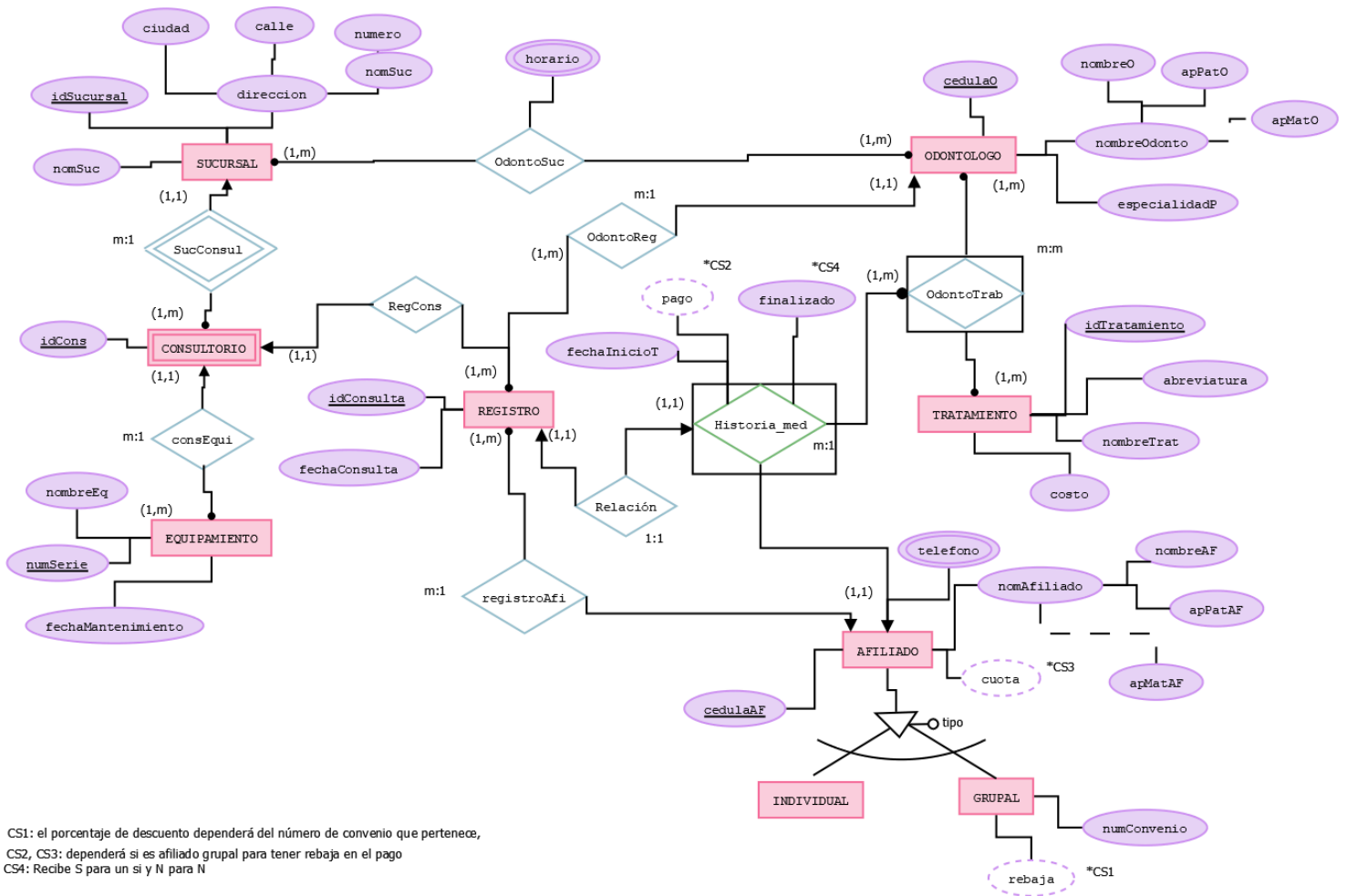
ÍNDICE

A. Modelo Entidad-Relación.....	2
B. Modelo Relacional.....	3
D. Creación de la estructura usando Oracle.....	5
E. Inserción de Datos usando INSERT INTO.....	8
F. Vista.....	14
G. Procedimientos.....	14
I. Inserciones a través de procedimientos.....	14
II. Actualizaciones a través de Procedimientos.....	23
III. Borrar.....	24
H. Actividades.....	25
1. Ejercicio 1.....	25
2. Ejercicio 2.....	27
4. Ejercicio 4.....	29
5. Ejercicio 5.....	30
6. Ejercicio 6.....	31
7. Ejercicio 7.....	31
8. Ejercicio 8.....	32
9. Ejercicio 9.....	33
10. Ejercicio 10.....	34

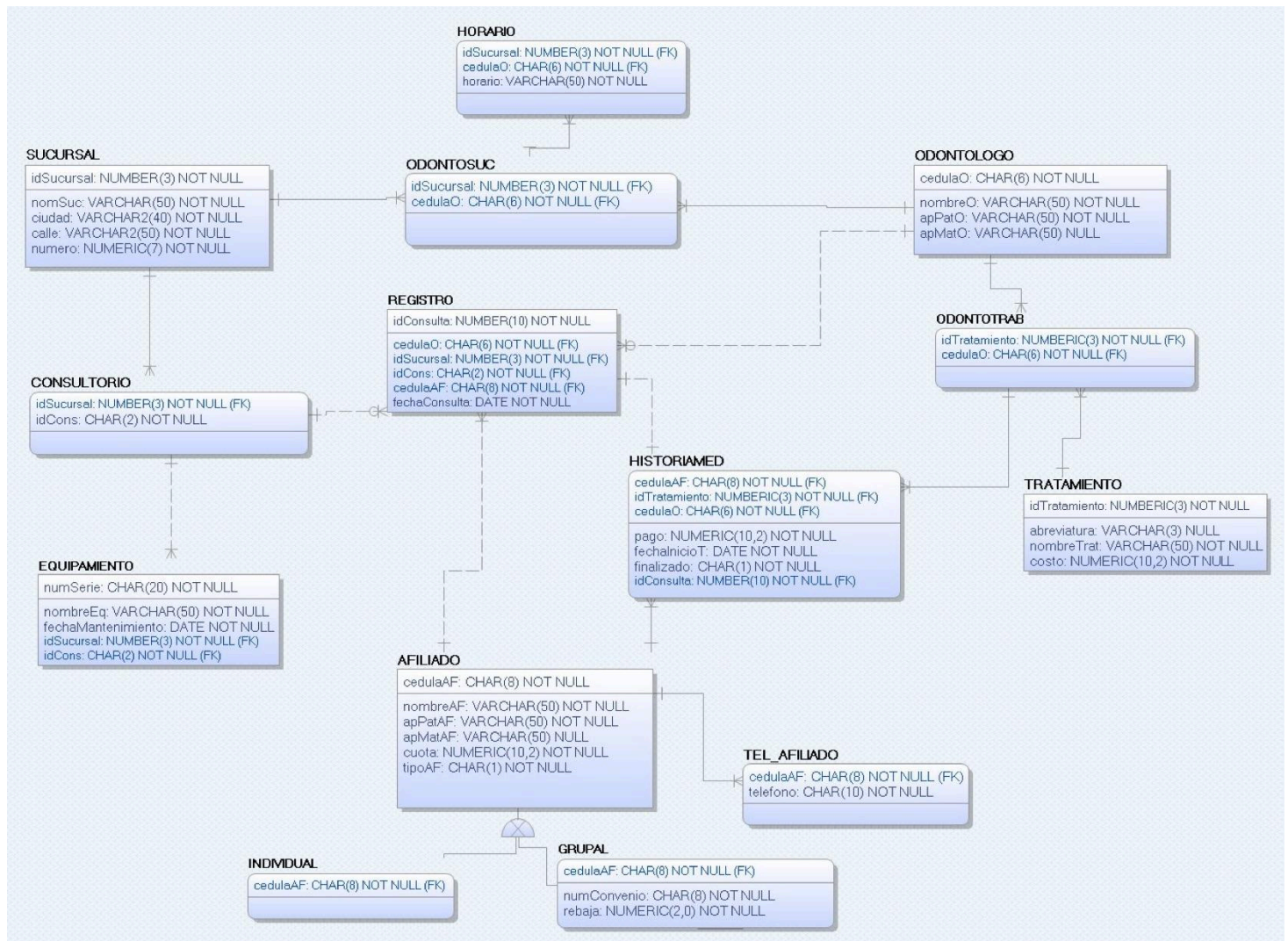
CONSULTORIO ODONTOLOGICO

(DISEÑO Y CONSTRUCCIÓN DE BASE DE DATOS)

A. Modelo Entidad-Relación



B. Modelo Relacional



C. Normalización

Para revisar que nuestras tablas estuvieran normalizadas optamos por crear prototipos de las tablas en excel para poder observar el comportamiento de los datos en ellas

SUCURSAL

SUCURSAL				
idSucursal	nomSuc	ciudad	calle	numero
1	Clínica Dental Sonrisa Perfecta	Madrid	Calle Mayor	45
2	Consultorio Dental Brillante	Barcelona	Avenida Diagonal	233
3	Centro Odontológico Diente de Oro	Valencia	Calle de la Paz	67
4	Odontología Avanzada Blanca	Sevilla	Calle Sierpes	102
5	Clínica Dental Salud y Estética	Bilbao	Gran Vía	180

La tabla sucursal tiene como llave primaria a idSucursal, se encuentra en primera forma normal ya que no posee atributos multivaluados, los datos son atomicos y no crece de forma horizontal; por otra parte se encuentra en segunda forma normal ya que todos los elementos dependen de nuestra llave principal idSucursal, se encuentra en 3FN ya que no hay dependencias internas, a primera vista se podría creer que entre ciudad y calle podría existir sin embargo esto no es posible ya que a pesar por cada ciudad no siempre va a ser la misma calle. No usamos normalización Boyce-Codd ya que no existe una llave compuesta y a pesar de que nomSuc es una candidata es demasiado larga para ser llave principal. Finalmente esta en cuarta forma normal ya que no tiene dependencias multivaluadas.

CONSULTORIO

CONSULTORIO	
idSucursal	idConsultorio
1	1
1	5
2	1
3	4
5	4
4	1
2	6
1	13
4	2
5	1
2	4
3	5

La tabla consultorio es una entidad debil que depende de Sucursal por lo cuál la llave principal es el id de la sucursal (idSucursal) y el id del consultorio (idCons) se encuentra en primera forma normal ya que en cada espacio los datos son atomicos y no crece de forma horizontal, se encuentra en segunda forma normal ya que al tener solamente dos atributos y estos son llave no existen elementos que no dependen de la llave principal, por lo mismo no hay dependencias internas con otros atributos, cumple forma normal de Boyce-Codd ya que no hay posibles llaves candidatas que puedan implicar una normalización Boyce-Codd. Finalmente esta en cuarta forma normal ya que no tiene dependencias multivaluadas.

EQUIPAMIENTO

EQUIPAMIENTO				
numSerie	nombreEq	fechaMantenimiento	idCons	idSuc
12354685296314785236	DentalPro Ultrasonic Scaler	4 de febrero de 2023	1	1
14896325148976432894	OrthoMax Digital X-Ray System	5 de noviembre de 2020	1	2
89631452789632145624	PerioClean Laser Therapy Unit	12 de abril de 2021	4	3
14785232145691735826	AeroDrill High-Speed Handpiece	23 de agosto de 2022	4	5
15975365485236987415	FlexiChair Ergonomic Dental Stool	31 de marzo de 2024	1	2

La tabla equipamiento tiene como llave primaria a numSerie, se encuentra en primera forma normal ya que todos los datos son atomicos, cumple segunda forma normal ya que todos los elementos dependen de nuestra llave principal completa, se encuentra en 3FN ya que no hay dependencias internas, podría aparentar que entre sucursal y consultorio podría existir una dependencia pero no es posible ya que el id del consultorio se puede repetir en otras sucursales y cada sucursal va tener muchos consultorios. No usamos normalización Boyce-Codd ya que no existe una llave compuesta y por lo tanto no hay candidatas que impliquen esta normalización. Finalmente esta en cuarta forma normal ya que no tiene dependencias multivaluadas.

ODONTOLOGO

ODONTOLOGO			
cedulaO	nombreO	apPatO	apMatO
12345678	Laura Sofía	Martínez	Gómez
23456789	Carlos Andrés	Ramírez	Pérez
34567890	María Fernanda	López	Rodríguez
45678901	Juan Pablo	Hernández	Sánchez
56789012	Ana Isabel	Torres	Morales

La tabla Odontologo tiene como llave principal a cedulaO, se encuentra en 1FN ya que no crece de forma horizontal y todos los datos son atomicos, se encuentra en 2FN ya que todos los elementos (en este caso el nombre completo) dependen por completo de la llave principal, esta en 3FN ya que no hay dependencias internas entre las demás columnas, no es necesario aplicar Boyce-Cod ya que no es una llave compuesta y no hay candidatas y por ultimo esta en 4FN ya que no hay dependencias multivaluadas.

ODONTOSUC

ODONTOSUC	
idSucursal	cedulaO
1	12345678
2	23456789
4	56789012
4	12345678
3	56789012

Esta tabla representa la relación entre el odontologo y la sucursal como su centro del trabajo, proviene de una relación muchos a muchos por lo cual la llave principal es el id de la sucursal con la cedula del odontologo, esta en primera forma normal ya que todos los datos son atomicos, al solo tener dos atributos y estos son llave primaria, no existe más elementos que no dependan de la llave principal, cumple tercera forma normal ya que al ser solo dos elementos no existe una transitividad, por la misma razón y sumando el hecho de que no hay llaves candidatas cumple forma normal de

Boyce- Cod, fianlemente tambien se encuentra en 4FN ya que no existen dependencias multivaluadas.

HORARIO

HORARIO		
idSucursal	cedulaO	Horario
1	12345678	Lun 2:00-4:40
2	23456789	Lun. Mar.Vie 12:00-13:00
4	56789012	Mar 15:00-16:00
4	12345678	Jue 15:00-16:00
3	56789012	Lun. Mar.Vie 12:00-13:00

Horario es una tabla proveniente del atributo multivaluado de la relacion entre el odontologo y la sucursal, creamos esta tabla para evitar que apliquemos posteriormente la 4FN, en esta tabla la cedula del odontologo, la sucursal y el horario son la llave primaria, esta en primera forma normal ya que solo existen valores atomicos en nuestra tabla y no crece horizontalmente, esta en 2FN ya que solo existen tres atributos que juntos son la llave principal por lo cual no hay atributos que no dependan de toda la llave, tampoco existe ytransividad por lo cual esta en 3FN, a l no tener llaves candidatas no e s necesario aplicar Boyce- Cod y fianalmente no tiene dependencias multivaluadas por lo cuál esta en 4FN

TRATAMIENTO

TRATAMIENTO			
idTratamiento	abreviatura	nombre	costo
1	BD	Blanqueamiento Dental	250
2	OB	Ortodoncia con Brackets	1500
3	LDP	Limpieza Dental Profesional	80
4	ID	Implante Dental	2000
5	CP	Carillas de Porcelana	900

Esta tabla tiene como llave principal al id del tratamiento, una llave artificial proveneinete de abreviatura y nombre que son llaves candidatas, esta en primer forma normal ya que no hay elementos o valores multivaluados, al ser una llave simple todos los elementos dependen por completo de ella, no existe algun tipo de dependencia interna ya que los tratamientos solo se representan pro su nombre y no posee alguna categoria que nos haga pensar que el nombre puede depender de ella, por otro lado la abreviatura y el nombre a pesar de que parecen similares representan cosas distintas ya que la abrteviatura nos representa la descripción del procedimiento. Esta en 4 forma normal ya que no hay dependencia multivaluadas.

ODONTOTRAB

ODONTRAB	
idTratamiento	cedulaO
5	23456789
5	45678901
4	23456789
2	45678901
4	56789012

Esta tabla es una relación muchos a muchos entre el odontologo y los tratamientos que es capaz de realizar por lo que la llave principal esta compuesta por la cedula del odontologo y el id del tratamiento, esta en 1FN ya que los datos son atomicos, al solo tener dos atributos y estos son llave primaria, no existen más elementos que no dependan de la llave principal, cumple tercera forma normal ya que no existe una transitividad, al no existir más elementos cumple Boyce-Codd y no posee dependencias multivaluadas (4FN).

AFILIADO

AFILIADO					
cedulaAF	nombreAF	apPatAF	apMatAF	cuota	tipoAF
87654321	Diego Alejandro	Fernández		500	I
98765432	Camila Valentina	García	Torres	800.63	G
76543210	José Luis	Martínez	Ramírez	600	G
87654322	Andrea Paola	López	Hernández	550	I
65432109	Juan Carlos	Sánchez	Pérez	650	I

La tabla afiliado tiene como llave principal al Id del afiliado, se encuentra en 1FN ya que no tiene elementos multivaluados ni crece horizontalmente, esta en 2FN ya que todos los atributos como su nombre y tipo dependen por completo de la llave principal, cuota en si es la misma para todos pero no por eso no depende de la llave principal y no existe algun elemento que dependa de la cuota. No existió necesidad de usar Boyce-Codd ya que no tenemos una llave compuesta, ni hay candidatas, por último cumple 4FN ya que no tiene dependencias multivaluadas ya que creamos una tabla extra para los telefonos de los afiliados.

TELEFONO

TEL_AFILIADO	
cedulaAF	telefono
98765432	5565132802
76543210	5569823512
65432109	5689232563
98765432	5674123984
76543210	5564128209

Esta tabla se genera a partir del afiliado, telefono era un atributo multivaluado del afiliado sin embargo para evitar que nuestra tabla no estuviera normalizada y siguiendo las reglas creamos una relación para ella. Esta en primera forma normal ya que no crece horizontalmente, ni tiene elementos multivaluados. Cumple segunda forma normal ya que existen unicamente dos elementos que son llave primaria por lo cual no hay elementos que no dependen por completo de la llave principal. Cumple tercera forma normal ya que no hay transitividad entre los atributos no llave en la relacion. Cumple Boyce Codd normal ya que se encuentra en tercera forma normal, y no hay posibles llaves candidatas, finalmente no hay dependencias multivaluadas para aplicar 4FN.

GRUPAL

Grupal		
cedulaAF	numConvenio	rebaja
87654321	12365478	25
65432109	14526398	17
87654322	17823956	58

La tabla grupal esta compuesta por la cedula del afiliado como llave principal y como atributos en numero de convenio y el porcentaje de rebaja. Se encuentra en 1FN ya que todos los datos que posee son atomicos, por ejemplo no vamos a tener dos porcentajes de rebaja ocupando el mismo registro, esta en segunda forma normal ya que solo existe una llave y los otros dos atributos dependen por completo de la llave, pudiendo repetirse el numero de convenio pero al final depende de la cedula del afiliado. Esta en tercera forma normal ya que no hay una dependencia transitive entre los dos atributos y la llave. No fue necesario usar Boyce Codd ya que no tenemos una llave compuesta y no hay posibles candidatas ya que como dijimos el numero de convenio se puede repetir. Finalmente esta en 4FN ya que ni cuota ni numero de convenio tienen una dependencia multivaluada.

REGISTRO

REGISTRO					
idConsulta	cedulaO	cedulaAF	fechaConsulta	idSucursal	idConsultorio
1	12345678	87654321	12-04-24	5	4
2	23456789	98765432	15-06-96	1	13
3	56789012	98765432	12-04-24	5	4
4	12345678	98765432	26-03-12	4	2
5	56789012	87654321	14-09-23	1	13

Esta tabla representa el conjunto de consultas otorgadas a un afiliado, la llave principal es la cedula del odontologo, la cedula del afiliado y la fecha de tal forma que no se repitan consultas pero si permita generar más con distintas características. Esta en primera forma normal ya que los datos en los registros son atomicos, es decir una consulta no tiene asignados 2 consultorios a la vez en el mismo registro. Esta en segunda forma normal ya que como podemos ver el pago no solo depende del odontologo, paciente o fecha sino que depende de los tres en conjunto, se encuentra en tercera forma normal ya que no existe una transitividad o dependencia interna entre ellos. No fue necesario aplicar normalización de Boyce-Codd ya que aunque a primera vista pareciera que el consultorio podría ser una llave candidata que sustituya a odontologo el consultorio también debería poder generar ya sea al afiliado o al odontólogo, lo cual no sucede, además si lo analizamos no tendría sentido que el consultorio fuera parte de la llave ya que nos causaría limitaciones. Finalmente está en 4 forma normal ya que no existe ninguna dependencia multivaluada.

HISTORIA_MED

HISTORIA_MED						
cedulaAF	lidTratamiento	cedulaO	pago	fechalnicio	finalizado	idConsulta
98765432	1	23456789	1200	12-04-24	N	2
98765432	3	56789012	5000	15-05-23	N	4
76543210	2	12345678	340.56	16-06-24	N	5
98765432	1	12345678	890	12-04-24	N	6
87654321	5	56789012	158	14-06-06	S	8
87654321	5	56789012	789.89	03-06-03	S	9

La tabla historia médica está conformada por la cédula del afiliado, la del odontólogo y el id del tratamiento, además posee los atributos de fecha de inicio, pago, status de finalizado y el id de la consulta que la generó. Está en primera forma normal ya que no existen elementos multivaluados, es decir a un id no le corresponden mas de un atributo en una misma casilla, esta en segunda forma normal ya que tanto el pago, la fecha de inicio, la consulta y el status de finalizado dependen de toda la llave completa ya que el pago no es sólo dado por el id del tratamiento o del paciente sino que de ambos, la fecha de inicio depende de los tres elementos, el status también. Está en 3FN ya que no existe ninguna dependencia transitiva entre los atributos, es decir no existe una relación por ejemplo entre la fecha de inicio y el monto total del pago o del status de finalizado con la fecha de inicio ya que este nos calcula. Tampoco es necesario aplicar Boyce Cod ya que no tenemos una llave candidata que nos pueda generar a otra ya que no tendría sentido y provocaría una falta de coherencia en la base de datos. Por último no existen dependencias multivaluadas, es decir que una misma llave le correspondan muchos valores de alguna columna, por lo cuál está en 4FN.

Una vez que realizamos nuestra normalización y comprobamos que no hay repetición de datos, falta de coherencia o algún otro problema, estamos listos para implementar nuestro diseño.

D. Creación de la estructura usando Oracle

SUCURSAL

```
CREATE TABLE sucursal(
idSucursal NUMBER(3) GENERATED BY DEFAULT AS IDENTITY CONSTRAINT PK_Sucursal PRIMARY KEY,
nomSuc VARCHAR2(50) NOT NULL CONSTRAINT AK_nomSuc UNIQUE,
ciudad VARCHAR2(40) NOT NULL,
calle VARCHAR2(50) NOT NULL,
numero NUMERIC(7) NOT NULL
);
```

CONSULTORIO

```
CREATE TABLE consultorio(
idSucursal NUMBER(3),
idCons CHAR(2),
CONSTRAINT PK_consultorio PRIMARY KEY(idSucursal,idCons),
CONSTRAINT FK_sucCons FOREIGN KEY(idSucursal)
REFERENCES sucursal ON DELETE CASCADE
);
```

EQUIPAMIENTO

```
CREATE TABLE equipamiento(
numSerie CHAR(20) CONSTRAINT PK_equipamiento PRIMARY KEY,
nombreEq VARCHAR2(50) NOT NULL,
```

```

fechaMantenimiento DATE NULL,
idSucursal NUMBER(3) NOT NULL,
idCons CHAR(2) NOT NULL,
CONSTRAINT FK_equipCons FOREIGN KEY(idSucursal,idCons)
REFERENCES consultorio ON DELETE SET NULL
);

```

ODONTOLOGO

```

CREATE TABLE odontologo (
cedulaO CHAR(6) CONSTRAINT PK_odontologo PRIMARY KEY,
nombreO VARCHAR2(50) NOT NULL,
apPatO VARCHAR2(50) NOT NULL,
apMatO VARCHAR2(50),
especialidadP VARCHAR(50) NOT NULL
);

```

ODONTOSUC

(Relación entre un odontólogo y su centro de trabajo)

```

CREATE TABLE odontoSuc (
idSucursal NUMBER(3),
cedulaO CHAR(6),
CONSTRAINT PK_odontoSuc PRIMARY KEY (idSucursal,cedulaO),
CONSTRAINT FK_sucOS FOREIGN KEY (idSucursal)
REFERENCES sucursal ON DELETE CASCADE,
CONSTRAINT FK_odontoOS FOREIGN KEY (cedulaO)
REFERENCES odontologo ON DELETE CASCADE
);

```

HORARIO

(Horario de trabajo en cada Sucursal de un Odontólogo)

```

CREATE TABLE horario(
idSucursal NUMBER(3),
cedulaO CHAR(6),
horario VARCHAR(50),
CONSTRAINT PK_horario PRIMARY KEY (cedulaO, horario),
CONSTRAINT FK_odontologoHor FOREIGN KEY(idSucursal, cedulaO)
REFERENCES odontoSuc ON DELETE CASCADE
);

```

TRATAMIENTO

```

CREATE TABLE tratamiento(
idTratamiento NUMBER(3) GENERATED BY DEFAULT AS IDENTITY
CONSTRAINT PK_tratamiento PRIMARY KEY,
abreviatura VARCHAR(3) NOT NULL CONSTRAINT AK_tratamiento UNIQUE,
nombreTrat VARCHAR(50) NOT NULL CONSTRAINT AK_nomtratamiento UNIQUE,
costo NUMERIC(10,2) NOT NULL
);

```

ODONTOTRAB

(Cátalo de tratamientos que un Odontólogo es capaz de realizar)

```
CREATE TABLE odontoTrab(  
idTratamiento NUMBER(3),  
cedulaO CHAR(6),  
CONSTRAINT PK_odontoTrab PRIMARY KEY(idTratamiento,cedulaO),  
CONSTRAINT FK_tratOT FOREIGN KEY (idTratamiento)  
REFERENCES tratamiento,  
CONSTRAINT FK_odontoOT FOREIGN KEY(cedulaO)  
REFERENCES odontologo  
);
```

AFILIADO

```
CREATE TABLE afiliado(  
cedulaAF CHAR(8) NOT NULL CONSTRAINT PK_afiliado PRIMARY KEY,  
nombreAF VARCHAR2(50) NOT NULL,  
apPatAF VARCHAR2(50) NOT NULL,  
apMatAF VARCHAR2(50),  
cuota NUMERIC(10,2) NOT NULL,  
tipoAF CHAR(1) NOT NULL,  
CONSTRAINT CK_tipoAF CHECK(tipoAF IN ('I','G'))  
);
```

TELEFONO

(Directorio de Telefonos para cada Afiliado)

```
CREATE TABLE telefono(  
cedulaAF CHAR(8),  
telefono CHAR(10),  
CONSTRAINT PK_telefonoAF PRIMARY KEY(cedulaAF,telefono),  
CONSTRAINT FK_afiliadoTel FOREIGN KEY(cedulaAF)  
REFERENCES afiliado ON DELETE CASCADE  
);
```

GRUPAL

(Tipo de afiliado cuyo contrato es grupal)

```
CREATE TABLE grupal(  
cedulaAF CHAR(8) NOT NULL CONSTRAINT PK_grupalAF PRIMARY KEY,  
numConvenio CHAR(8) NOT NULL,  
rebaja NUMERIC(2,0) NOT NULL,  
CONSTRAINT FK_afiliadoGrp FOREIGN KEY(cedulaAF)  
REFERENCES afiliado ON DELETE CASCADE  
);
```

REGISTRO

(Registros de las consultas tomadas para los Afiliados)

```
CREATE TABLE registro(  
idConsulta NUMBER(10) GENERATED ALWAYS AS IDENTITY,
```

```

cedulaO CHAR(6) NOT NULL,
idSucursal NUMBER(3) NOT NULL,
idCons CHAR(2) NOT NULL,
cedulaAF CHAR(8) NOT NULL,
fechaConsulta DATE DEFAULT SYSDATE NOT NULL,
CONSTRAINT PK_consulta PRIMARY KEY(idConsulta),
CONSTRAINT FK_odontologoOC FOREIGN KEY(cedulaO)
REFERENCES odontologo,
CONSTRAINT FK_consultorioOC FOREIGN KEY(idSucursal, idCons)
REFERENCES consultorio,
CONSTRAINT FK_afiliadoOC FOREIGN KEY(cedulaAF)
REFERENCES afiliado,
CONSTRAINT FK_odontoSuc FOREIGN KEY(idSucursal,cedulaO)
REFERENCES odontoSuc,
CONSTRAINT AK_registro UNIQUE(cedulaO, cedulaAF, fechaConsulta)
);

```

HISTORIA_MED

(Historia Médica de los tratamientos practicados a los Afiliados)

```

CREATE TABLE historia_Med(
cedulaAF CHAR(8),
idTratamiento NUMBER(3),
cedulaO CHAR(6),
pago NUMBER(10,2) NOT NULL,
fechaInicioT DATE NOT NULL,
finalizado CHAR(1) NOT NULL,
idConsulta NUMBER(10) NOT NULL,
CONSTRAINT PK_historialMed PRIMARY KEY(cedulaAF,idTratamiento, cedulaO),
CONSTRAINT FK_tratamientoHM FOREIGN KEY(idTratamiento,cedulaO)
REFERENCES odontoTrab,
CONSTRAINT FK_afiliadoHM FOREIGN KEY(cedulaAF)
REFERENCES afiliado,
CONSTRAINT fk_consulta FOREIGN KEY (idConsulta)
REFERENCES registro,
CONSTRAINT CK_statusFinalizado CHECK (finalizado IN ('S','N'))
);

```

E. Inserción de Datos usando INSERT INTO.

SUCURSAL

```

INSERT INTO sucursal(nomSuc,ciudad,calle,numero)
VALUES ('Clínica Dental Sonrisa Perfecta','Madrid','Calle Mayor',45);

INSERT INTO sucursal(nomSuc,ciudad,calle,numero)
VALUES ('Centro Odontológico Diente de Oro','Valencia','Calle de la Paz',67);

INSERT INTO sucursal(nomSuc,ciudad,calle,numero)
VALUES('Consultorio Dental Brillante','Barcelona','Avenida Diagonal',233);

INSERT INTO sucursal(nomSuc,ciudad,calle,numero)
VALUES('Odontología Avanzada Blanca','Sevilla','Calle Sierpes',102);

```

```

INSERT INTO sucursal(nomSuc,ciudad,calle,numero)
VALUES('Clínica Dental Salud y Estética','Bilbao','Gran Vía',180);

INSERT INTO sucursal(nomSuc,ciudad,calle,numero)
VALUES('Clínica Dental Sonrisa Radiante','Buenos Aires','Avenida Corrientes',1720);

INSERT INTO sucursal(nomSuc,ciudad,calle,numero)
VALUES('Odontología Integral SaludBucal','Lima','Calle Las Begonias',340);

INSERT INTO sucursal(nomSuc,ciudad,calle,numero)
VALUES('Centro Odontológico Dentisano','Santiago','Calle Huérfanos',995);

INSERT INTO sucursal(nomSuc,ciudad,calle,numero)
VALUES('Consultorio Dental Smiles','Bogotá','Carrera 15',45);

INSERT INTO sucursal(nomSuc,ciudad,calle,numero)
VALUES('Clínica Dental Bienestar Oral','Ciudad de México','Paseo de la Reforma',230);

```

CONSULTORIO

```

INSERT INTO consultorio(idSucursal,idCons) VALUES(1,'A1');
INSERT INTO consultorio(idSucursal,idCons) VALUES(2,'D1');
INSERT INTO consultorio(idSucursal,idCons) VALUES(4,'A1');
INSERT INTO consultorio(idSucursal,idCons) VALUES(4,'C1');
INSERT INTO consultorio(idSucursal,idCons) VALUES(5,'B1');
INSERT INTO consultorio(idSucursal,idCons) VALUES(6,'F2');
INSERT INTO consultorio(idSucursal,idCons) VALUES(8,'G7');
INSERT INTO consultorio(idSucursal,idCons) VALUES(8,'87');
INSERT INTO consultorio(idSucursal,idCons) VALUES(10,'D7');
INSERT INTO consultorio(idSucursal,idCons) VALUES(6,'D7');

```

EQUIPAMIENTO

```

INSERT INTO equipamiento(numSerie,nombreEq,fechaMantenimiento,idSucursal,idCons)
VALUES('USP2024SCALERPRO01','Ultrasonic Scaler Pro','15-03-2024',10,'D7');

INSERT INTO equipamiento(numSerie,nombreEq,fechaMantenimiento,idSucursal,idCons)
VALUES('DXM2024XRAYMACH02','Digital X-Ray Machine','22-04-24',1,'A1');

INSERT INTO equipamiento(numSerie,nombreEq,fechaMantenimiento,idSucursal,idCons)
VALUES('HSH2024HSPDHPCE03','High-Speed Handpiece','10-05-24',6,'F2');

INSERT INTO equipamiento(numSerie,nombreEq,fechaMantenimiento,idSucursal,idCons)
VALUES('DA2024AUTOCLAVE04','Dental Autoclave','01-06-23',2,'D1');

INSERT INTO equipamiento(numSerie,nombreEq,fechaMantenimiento,idSucursal,idCons)
VALUES('IC2024INTRAORAL05','Intraoral Camera','18-06-20',4,'A1');

INSERT INTO equipamiento(numSerie,nombreEq,fechaMantenimiento,idSucursal,idCons)
VALUES('DCD2024CHAIRDLX06','Dental Chair Deluxe','30-08-21',4,'C1');

INSERT INTO equipamiento(numSerie,nombreEq,fechaMantenimiento,idSucursal,idCons)
VALUES('LCL2024CURINGLGT07','LED Curing Ligh','12-09-10',5,'B1');

INSERT INTO equipamiento(numSerie,nombreEq,fechaMantenimiento,idSucursal,idCons)
VALUES('AL2024APEXLOCTR08','Apex Locator','24-10-22',8,'G7');

INSERT INTO equipamiento(numSerie,nombreEq,fechaMantenimiento,idSucursal,idCons)
VALUES('DSU2024SUCTIONU09','Dental Suction Unit','05-11-09',10,'D7');

```

```
INSERT INTO equipamiento(numSerie,nombreEq,fechaMantenimiento,idSucursal,idCons)
VALUES('PDU2024PORTUNIT10','Portable Dental Unit','17-12-21',6,'D7');
```

ODONTOLOGO

```
INSERT INTO odontologo (cedulaO, nombreO, appatO, apmatO, especialidadP)
VALUES ('123456','Ana María','Rodríguez','Lopez','Ortodoncia');
```

```
INSERT INTO odontologo (cedulaO, nombreO, appatO, apmatO, especialidadP)
VALUES ('654321','Juan Carlos','Pérez','Martínez','Prostodoncia');
```

```
INSERT INTO odontologo (cedulaO, nombreO, appatO, apmatO, especialidadP)
VALUES ('112233','Laura','Gómez','Fernández','Periodoncia');
```

```
INSERT INTO odontologo (cedulaO, nombreO, appatO, apmatO, especialidadP)
VALUES ('445566','Luis','Herrera','García','Odontopediatria');
```

```
INSERT INTO odontologo (cedulaO, nombreO, appatO, apmatO, especialidadP)
VALUES ('778899','Carmen Elena','Ramírez','Torres','Cirugia Oral y Maxilofacial');
```

```
INSERT INTO odontologo (cedulaO, nombreO, appatO, apmatO, especialidadP)
VALUES ('223344','Pedro José','Ruiz','Gonzalez','Ortodoncia' );
```

```
INSERT INTO odontologo (cedulaO, nombreO, appatO, apmatO, especialidadP)
VALUES ('556677','María Fernanda','Ortega','Sanchez','Odontologia Estetica');
```

```
INSERT INTO odontologo (cedulaO, nombreO, appatO, apmatO, especialidadP)
VALUES ('998877','Francisco','Morales','Diaz','Prostodoncista');
```

```
INSERT INTO odontologo (cedulaO, nombreO, appatO, apmatO, especialidadP)
VALUES ('334455','Patricia','Castillo','Vega','Odontologia Restauradora');
```

```
INSERT INTO odontologo (cedulaO, nombreO, appatO, apmatO, especialidadP)
VALUES ('889900','Carmen Elena','Ramírez','Torres','Cirugia Oral y Maxilofacial');
```

TRATAMIENTO

```
INSERT INTO tratamiento (abreviatura, nombreTrat, costo)
VALUES ( 'OCB', 'Ortodoncia con Brackets', 1000);
```

```
INSERT INTO tratamiento (abreviatura, nombreTrat, costo)
VALUES ( 'BD', 'Blanqueamiento Dental ', 2500);
```

```
INSERT INTO tratamiento (abreviatura, nombreTrat, costo)
VALUES ( 'LDP', 'Limpieza Dental Profesional', 1500);
```

```
INSERT INTO tratamiento (abreviatura, nombreTrat, costo)
VALUES ( 'ID', 'Implante Dental', 12000);
```

```
INSERT INTO tratamiento (abreviatura, nombreTrat, costo)
VALUES ( 'CP', 'Carillas de Porcelana', 900);
```

```
INSERT INTO tratamiento (abreviatura, nombreTrat, costo)
VALUES ( 'ED','Extracción de Dientes', 2500);
```

```
INSERT INTO tratamiento (abreviatura, nombreTrat, costo)
VALUES ( 'PE', 'Periodoncia', 450);
```

```
INSERT INTO tratamiento (abreviatura, nombreTrat, costo)
VALUES ( 'SD', 'Selladores Dentales:', 600);
```



```
INSERT INTO tratamiento (abreviatura, nombreTrat, costo)
VALUES ( 'INC', 'Incrustaciones',600);
```

```
INSERT INTO tratamiento (abreviatura, nombreTrat, costo)
VALUES ( 'CO', 'Cirugía Ortognática',30000 );
```

ODONTOSUC

```
INSERT INTO odontoSuc(idSucursal,cedulaO) VALUES(1,'889900');
INSERT INTO odontoSuc(idSucursal,cedulaO) VALUES(2,'654321');
INSERT INTO odontoSuc(idSucursal,cedulaO) VALUES(3,'778899');
INSERT INTO odontoSuc(idSucursal,cedulaO) VALUES(3,'998877');
INSERT INTO odontoSuc(idSucursal,cedulaO) VALUES(4,'778899');
INSERT INTO odontoSuc(idSucursal,cedulaO) VALUES(4,'889900');
INSERT INTO odontoSuc(idSucursal,cedulaO) VALUES(4,'654321');
INSERT INTO odontoSuc(idSucursal,cedulaO) VALUES(8,'998877');
INSERT INTO odontoSuc(idSucursal,cedulaO)VALUES(8,'112233');
INSERT INTO odontoSuc(idSucursal,cedulaO)VALUES(8,'778899');
```

HORARIO

```
INSERT INTO horario (idSucursal,cedulaO, horario) VALUES (1,'889900','lun,mar de 7:00-13:00');
INSERT INTO horario (idSucursal,cedulaO, horario) VALUES (1,'889900','mie de 10:00-14:00');
INSERT INTO horario (idSucursal,cedulaO, horario) VALUES (2,'654321','lun,mar,jue de 16:00-20:00');
INSERT INTO horario (idSucursal,cedulaO, horario) VALUES (2,'654321','vie,sab de 7:00-14:00');
INSERT INTO horario (idSucursal,cedulaO, horario) VALUES (3,'778899','lun,mar de 13:00-22:00');
INSERT INTO horario (idSucursal,cedulaO, horario) VALUES (3,'998877','mie,jue,vie de 12:00-19:00');
INSERT INTO horario (idSucursal,cedulaO, horario) VALUES (4,'778899','mar, jue de 8:00-12:00');
INSERT INTO horario (idSucursal,cedulaO, horario) VALUES (4,'778899','lun, mie de 8:00-14:00');
INSERT INTO horario (idSucursal,cedulaO, horario) VALUES (4,'889900','vie,sab,dom de 12:00-17:00');
INSERT INTO horario (idSucursal,cedulaO, horario) VALUES (4,'654321','jue de 15:00-20:00');
INSERT INTO horario (idSucursal,cedulaO, horario) VALUES (8,'998877','lun, vie de 10:00-16:00');
INSERT INTO horario (idSucursal,cedulaO, horario) VALUES (8,'778899','mar,mier de 7:00-14:00');
INSERT INTO horario (idSucursal,cedulaO, horario) VALUES (8,'112233','sab de 11:00-17:00');
```

ODONTOTRAB

```
INSERT INTO odontoTrab (idTratamiento, cedulaO) VALUES (1, '123456');
INSERT INTO odontoTrab (idTratamiento, cedulaO) VALUES (2, '123456');
INSERT INTO odontoTrab (idTratamiento, cedulaO) VALUES (3, '123456');
INSERT INTO odontoTrab (idTratamiento, cedulaO) VALUES (4, '654321');
INSERT INTO odontoTrab (idTratamiento, cedulaO) VALUES (5, '654321');
INSERT INTO odontoTrab (idTratamiento, cedulaO) VALUES (6, '112233');
INSERT INTO odontoTrab (idTratamiento, cedulaO) VALUES (7, '112233');
INSERT INTO odontoTrab (idTratamiento, cedulaO) VALUES (8, '445566');
INSERT INTO odontoTrab (idTratamiento, cedulaO) VALUES (9, '445566');
INSERT INTO odontoTrab (idTratamiento, cedulaO) VALUES (10, '778899');
INSERT INTO odontoTrab (idTratamiento, cedulaO) VALUES (1, '223344');
INSERT INTO odontoTrab (idTratamiento, cedulaO) VALUES (3, '223344');
INSERT INTO odontoTrab (idTratamiento, cedulaO) VALUES (2, '556677');
INSERT INTO odontoTrab (idTratamiento, cedulaO) VALUES (5, '556677');
INSERT INTO odontoTrab (idTratamiento, cedulaO) VALUES (8, '556677');
INSERT INTO odontoTrab (idTratamiento, cedulaO) VALUES (4, '998877');
```

```

INSERT INTO odontoTrab (idTratamiento, cedulaO) VALUES (6, '998877');
INSERT INTO odontoTrab (idTratamiento, cedulaO) VALUES (7, '334455');
INSERT INTO odontoTrab (idTratamiento, cedulaO) VALUES (9, '334455');
INSERT INTO odontoTrab (idTratamiento, cedulaO) VALUES (10, '889900');

```

AFILIADO

```

INSERT INTO afiliado (cedulaAF, nombreAF, apPatAF, apMatAF, cuota, tipoAF)
VALUES ('GOMJ8501', 'Juan', 'Gómez', 'Martínez', 2000, 'I');

INSERT INTO afiliado (cedulaAF, nombreAF, apPatAF, apMatAF, cuota, tipoAF)
VALUES ('LOPF7902', 'Fernanda', 'López', 'Pérez', 2000, 'I');

INSERT INTO afiliado (cedulaAF, nombreAF, apPatAF, apMatAF, cuota, tipoAF)
VALUES ('HERS8903', 'Sofía', 'Hernández', 'Ruiz', 2000, 'I');

INSERT INTO afiliado (cedulaAF, nombreAF, apPatAF, apMatAF, cuota, tipoAF)
VALUES ('GARR7405', 'Roberto', 'García', 'Ramírez', 2000, 'I');

INSERT INTO afiliado (cedulaAF, nombreAF, apPatAF, apMatAF, cuota, tipoAF)
VALUES ('80', 'Raquel', 'Torres', 'Rojas', 2000, 'I');

INSERT INTO afiliado (cedulaAF, nombreAF, apPatAF, apMatAF, cuota, tipoAF)
VALUES ('SACM7707', 'Miguel', 'Sánchez', 'Castillo', 2000, 'G');

INSERT INTO afiliado (cedulaAF, nombreAF, apPatAF, apMatAF, cuota, tipoAF)
VALUES ('MORR8608', 'Ramón', 'Moreno', 'Romero', 2000, 'G');

INSERT INTO afiliado (cedulaAF, nombreAF, apPatAF, apMatAF, cuota, tipoAF)
VALUES ('JIMR7910', 'Rebeca', 'Jiménez', 'Martínez', 2000, 'G');

INSERT INTO afiliado (cedulaAF, nombreAF, apPatAF, apMatAF, cuota, tipoAF)
VALUES ('ALVM8001', 'Víctor', 'Álvarez', 'Muñoz', 2000, 'G');

INSERT INTO afiliado (cedulaAF, nombreAF, apPatAF, apMatAF, cuota, tipoAF)
VALUES ('RIVG8102', 'Gabriela', 'Rivera', 'Guzmán', 2000, 'G');

INSERT INTO afiliado (cedulaAF, nombreAF, apPatAF, apMatAF, cuota, tipoAF)
VALUES ('RODH8203', 'Héctor', 'Rodríguez', 'Hernández', 2000, 'G');

INSERT INTO afiliado (cedulaAF, nombreAF, apPatAF, apMatAF, cuota, tipoAF)
VALUES ('MONP8304', 'Patricia', 'Montes', 'Pérez', 2000, 'G');

INSERT INTO afiliado (cedulaAF, nombreAF, apPatAF, apMatAF, cuota, tipoAF)
VALUES ('CALR8405', 'Ricardo', 'Calderón', 'Lara', 2000, 'G');

INSERT INTO afiliado (cedulaAF, nombreAF, apPatAF, apMatAF, cuota, tipoAF)
VALUES ('MARM8506', 'Mónica', 'Martínez', 'Mendoza', 2000, 'G');

INSERT INTO afiliado (cedulaAF, nombreAF, apPatAF, apMatAF, cuota, tipoAF)
VALUES ('VALC8607', 'Carlos', 'Valdés', 'Cárdenas', 2000, 'G');

```

TELEFONOS

```

INSERT INTO telefono (cedulaAF, telefono) VALUES ('GOMJ8501', '5512345678');
INSERT INTO telefono (cedulaAF, telefono) VALUES ('LOPF7902', '5587654321');
INSERT INTO telefono (cedulaAF, telefono) VALUES ('HERS8903', '5512349876');
INSERT INTO telefono (cedulaAF, telefono) VALUES ('GARR7405', '5598765432');
INSERT INTO telefono (cedulaAF, telefono) VALUES ('SACM7707', '5598761234');
INSERT INTO telefono (cedulaAF, telefono) VALUES ('SACM7707', '5587659876');

```

```

INSERT INTO telefono (cedulaAF, telefono) VALUES ('MORR8608', '5512347654');
INSERT INTO telefono (cedulaAF, telefono) VALUES ('MORR8608', '5598765678');
INSERT INTO telefono (cedulaAF, telefono) VALUES ('JIMR7910', '5587651234');
INSERT INTO telefono (cedulaAF, telefono) VALUES ('JIMR7910', '5512349012');
INSERT INTO telefono (cedulaAF, telefono) VALUES ('ALVM8001', '5598765678');
INSERT INTO telefono (cedulaAF, telefono) VALUES ('ALVM8001', '5587659012');
INSERT INTO telefono (cedulaAF, telefono) VALUES ('ALVM8001', '5512347890');
INSERT INTO telefono (cedulaAF, telefono) VALUES ('RIVG8102', '5598761234');
INSERT INTO telefono (cedulaAF, telefono) VALUES ('RIVG8102', '5587658901');
INSERT INTO telefono (cedulaAF, telefono) VALUES ('RIVG8102', '5512345612');
INSERT INTO telefono (cedulaAF, telefono) VALUES ('RODH8203', '5598765123');
INSERT INTO telefono (cedulaAF, telefono) VALUES ('MONP8304', '5587651209');
INSERT INTO telefono (cedulaAF, telefono) VALUES ('CALR8405', '5512346789');
INSERT INTO telefono (cedulaAF, telefono) VALUES ('MARM8506', '5598760123');
INSERT INTO telefono (cedulaAF, telefono) VALUES ('VALC8607', '5587654567');

```

GRUPAL

```

INSERT INTO grupal (cedulaAF, numConvenio, rebaja) VALUES ('SACM7707', 'CON00001', 25);
INSERT INTO grupal (cedulaAF, numConvenio, rebaja) VALUES ('MORR8608', 'CON00001', 25);
INSERT INTO grupal (cedulaAF, numConvenio, rebaja) VALUES ('JIMR7910', 'CON00001', 25);
INSERT INTO grupal (cedulaAF, numConvenio, rebaja) VALUES ('ALVM8001', 'CON00001', 25);
INSERT INTO grupal (cedulaAF, numConvenio, rebaja) VALUES ('RIVG8102', 'CON00001', 25);
INSERT INTO grupal (cedulaAF, numConvenio, rebaja) VALUES ('RODH8203', 'CON00002', 15);
INSERT INTO grupal (cedulaAF, numConvenio, rebaja) VALUES ('MONP8304', 'CON00002', 15);
INSERT INTO grupal (cedulaAF, numConvenio, rebaja) VALUES ('CALR8405', 'CON00002', 15);
INSERT INTO grupal (cedulaAF, numConvenio, rebaja) VALUES ('MARM8506', 'CON00003', 10);
INSERT INTO grupal (cedulaAF, numConvenio, rebaja) VALUES ('VALC8607', 'CON00003', 10);

```

REGISTRO

```

INSERT INTO registro(cedulaO,idSucursal,idCons,cedulaAF,fechaConsulta)
VALUES('889900',1,'A1','GOMJ8501','12-05-24');

INSERT INTO registro(cedulaO,idSucursal,idCons,cedulaAF,fechaConsulta)
VALUES('112233',8,'87','GOMJ8501','22-11-22');

INSERT INTO registro(cedulaO,idSucursal,idCons,cedulaAF,fechaConsulta)
VALUES('998877',8,'87','MORR8608','02-10-20');

INSERT INTO registro(cedulaO,idSucursal,idCons,cedulaAF,fechaConsulta)
VALUES('654321',4,'C1','MORR8608','02-10-20');

INSERT INTO registro(cedulaO,idSucursal,idCons,cedulaAF,fechaConsulta)
VALUES('778899',4,'C1','ALVM8001','07-03-21');

INSERT INTO registro(cedulaO,idSucursal,idCons,cedulaAF,fechaConsulta)
VALUES('654321',2,'D1','RODH8203','08-12-08');

INSERT INTO registro(cedulaO,idSucursal,idCons,cedulaAF,fechaConsulta)
VALUES('778899',4,'C1','CALR8405','03-06-03');

INSERT INTO registro(cedulaO,idSucursal,idCons,cedulaAF,fechaConsulta)
VALUES('654321',2,'D1','MARM8506','14-06-06');

INSERT INTO registro(cedulaO,idSucursal,idCons,cedulaAF,fechaConsulta)
VALUES('889900',1,'A1','HERS8903','14-07-21');

```

```
INSERT INTO registro(cedulaO,idSucursal,idCons,cedulaAF,fechaConsulta)
VALUES('112233',8,'G7','RIVG8102','11-11-21');
```

HISTORIAL

```
INSERT INTO historia_Med (cedulaAF, idTratamiento, cedulaO, pago, fechaInicioT, finalizado,
idConsulta)
VALUES ('GOMJ8501', 10, '889900',1000, TO_DATE('2024-05-12', 'YYYY-MM-DD'), 'N', 1);

INSERT INTO historia_Med (cedulaAF, idTratamiento, cedulaO,pago, fechaInicioT, finalizado,
idConsulta)
VALUES ('GOMJ8501', 6, '112233', 2000, TO_DATE('2022-11-22', 'YYYY-MM-DD'), 'S', 2);
INSERT INTO historia_Med (cedulaAF, idTratamiento, cedulaO, pago, fechaInicioT, finalizado,
idConsulta)
VALUES ('MORR8608', 4, '998877', 12000, TO_DATE('2020-10-02', 'YYYY-MM-DD'), 'S', 3);
INSERT INTO historia_Med (cedulaAF, idTratamiento, cedulaO, pago, fechaInicioT, finalizado,
idConsulta)
VALUES ('MORR8608', 4, '654321', 6000, TO_DATE('2020-10-02', 'YYYY-MM-DD'), 'S', 4);
INSERT INTO historia_Med (cedulaAF, idTratamiento, cedulaO, pago, fechaInicioT, finalizado,
idConsulta)
VALUES ('ALVM8001', 10, '778899', 20000, TO_DATE('2021-03-07', 'YYYY-MM-DD'), 'N', 5);
INSERT INTO historia_Med (cedulaAF, idTratamiento, cedulaO, pago, fechaInicioT, finalizado,
idConsulta)
VALUES ('RODH8203', 4, '654321', 10000, TO_DATE('2008-12-08', 'YYYY-MM-DD'), 'S', 6);
INSERT INTO historia_Med (cedulaAF, idTratamiento, cedulaO, pago, fechaInicioT, finalizado,
idConsulta)
VALUES ('CALR8405', 10, '778899', 28000, TO_DATE('2003-06-03', 'YYYY-MM-DD'), 'S', 7);
INSERT INTO historia_Med (cedulaAF, idTratamiento, cedulaO, pago, fechaInicioT, finalizado,
idConsulta)
VALUES ('MARM8506', 4, '654321', 8000, TO_DATE('2006-06-14', 'YYYY-MM-DD'), 'S', 8);
INSERT INTO historia_Med (cedulaAF, idTratamiento, cedulaO, pago, fechaInicioT, finalizado,
idConsulta)
VALUES ('HERS8903', 10, '889900', 30000, TO_DATE('2021-07-14', 'YYYY-MM-DD'), 'N', 9);
INSERT INTO historia_Med (cedulaAF, idTratamiento, cedulaO, pago, fechaInicioT, finalizado,
idConsulta)
VALUES ('RIVG8102', 7, '112233', 400.00, TO_DATE('2021-11-11', 'YYYY-MM-DD'), 'S', 10);
```

F. Vista

Para el historial completo con consultas y tratamientos.

```
CREATE OR REPLACE VIEW vwhistorial AS
SELECT r.cedulaAF afiliado, r.cedulaO odonto, s.nomSuc sucursal, r.idcons consultorio,
r.fechaConsulta,
t.nombreTrat tratamiento, h.pago, h.fechaInicioT fecha_inicio, h.finalizado
FROM registro r
LEFT JOIN historia_Med h ON r.idConsulta=h.idConsulta
LEFT JOIN sucursal s ON s.idsucursal=r.idsucursal
LEFT JOIN tratamiento t ON t.idtratamiento=h.idtratamiento;
```

```
SQL> select * from vwhistorial;
```

AFILIADO	ODONTO	SUCURSAL	CO	FECHACON	TRATAMIENTO	PAGO	FECHA_IN	F
RODH8203	654321	Centro Odontológico Diente de Oro	D1	08/12/08	Implante Dental	10000	08/12/08	S
MARM8506	654321	Centro Odontológico Diente de Oro	D1	14/06/06	Implante Dental	8000	14/06/06	S
MORR8608	654321	Odontología Avanzada Blanca	C1	02/10/20	Implante Dental	6000	02/10/20	S
MORR8608	998877	Centro Odontológico Dentisano	S7	02/10/20	Implante Dental	12000	02/10/20	S
GOMJ8501	112233	Centro Odontológico Dentisano	S7	22/11/22	Extracción de Dientes	2000	22/11/22	S
RIVG8102	112233	Centro Odontológico Dentisano	G7	11/11/21	Periodoncia	400	11/11/21	S
GOMJ8501	889900	Clinica Dental Sonrisa Perfecta	A1	12/05/24	Cirugía Ortognática	1000	12/05/24	N
HERS8903	889900	Clinica Dental Sonrisa Perfecta	A1	14/07/21	Cirugía Ortognática	30000	14/07/21	N
ALVM8001	778899	Odontología Avanzada Blanca	C1	07/03/21	Cirugía Ortognática	20000	07/03/21	N
CALR8405	778899	Odontología Avanzada Blanca	C1	03/06/03	Cirugía Ortognática	28000	03/06/03	S

10 filas seleccionadas.

Para los afiliados que son individuales.

CREATE OR REPLACE VIEW individual AS

SELECT * FROM afiliado

WHERE tipoAF='I';

```
SQL> select * from individual;
```

CEDULAAF	NOMBREAF	APPATAF	APMTAF	CUOTA T
GOMJ8501	Juan	Gómez	Martínez	20
LOPE7902	Fernanda	López	Pérez	20
HERS8903	Sofía	Hernández	Ruiz	20
GARR7405	Roberto	García	Ramírez	20
	Raquel	Torres	Rojas	20

G. Procedimientos

I. Inserciones a través de procedimientos

SUCURSAL (SPaddSucursal)

```
CREATE OR REPLACE PROCEDURE SPaddSucursal(
vNomSuc IN VARCHAR2,
vCiudad IN VARCHAR2,
vCalle IN VARCHAR2,
vNumero IN NUMERIC)
AS
vSuc sucursal.idSucursal%TYPE;
BEGIN
SELECT idSucursal
INTO vSuc
FROM sucursal
WHERE nomSuc=vNomSuc;
IF (vSuc IS NOT NULL) THEN
DBMS_OUTPUT.PUT_LINE('Ya existe esta sucursal');
RAISE_APPLICATION_ERROR(-20001,'Ya existe esta sucursal');
END IF;
EXCEPTION
WHEN NO_DATA_FOUND THEN
INSERT INTO sucursal(nomSuc,ciudad,calle,numero)
VALUES(vNomSuc,vCiudad,vCalle,vNumero);
COMMIT;
DBMS_OUTPUT.PUT_LINE('Sucursal Creada' || vNomSuc);
END SPaddSucursal;
/
```

```

EXEC SPAddSucursal('Clinica Dental Reluciente', 'Monterrey', 'Avenida Garcia', 333);
EXEC SPAddSucursal('Odontología Vanguardista', 'Monterrey', 'Calle Hidalgo', 567);
EXEC SPAddSucursal('Clínica Dental Armonía Bucal', 'Guadalajara', 'Avenida Revolución', 234);
EXEC SPAddSucursal('Centro Odontológico Sonrisa Divina', 'Querétaro', 'Calle Zaragoza', 890);
EXEC SPAddSucursal('Consultorio Dental Boca Saludable', 'Cancún', 'Avenida Tulum', 678);
EXEC SPAddSucursal('Odontología Integral Dientes Felices', 'Tijuana', 'Bulevar Insurgentes', 345);
EXEC SPAddSucursal('Clínica Dental Brillo Dental', 'Puebla', 'Avenida Reforma', 456);
EXEC SPAddSucursal('Centro Odontológico Sonrisa Armoniosa', 'Mérida', 'Calle 59', 789);
EXEC SPAddSucursal('Consultorio Dental Dientes Sanos y Bellos', 'Hermosillo', 'Bulevar Kino', 234);
EXEC SPAddSucursal('Odontología Avanzada Salud Dental', 'San Luis Potosí', 'Avenida Universidad', 567);
EXEC SPAddSucursal('Clínica Dental Bucal Integral', 'Culiacán', 'Calle Constitución', 890);

```

IDSUCURSAL NUMERO	NOMSUC	CIUDAD	CALLE
567	12 Odontología Vanguardista	Monterrey	Calle Hidalgo
234	13 Clínica Dental Armonía Bucal	Guadalajara	Avenida Revolución
890	14 Centro Odontológico Sonrisa Divina	Querétaro	Calle Zaragoza
678	15 Consultorio Dental Boca Saludable	Cancún	Avenida Tulum
345	16 Odontología Integral Dientes Felices	Tijuana	Bulevar Insurgentes

CONSULTORIO

```

CREATE OR REPLACE PROCEDURE spAddConsultorio(
vNomSuc IN VARCHAR2,
vConsultorio IN VARCHAR2)
AS
vSuc sucursal.idSucursal%TYPE;
BEGIN
SELECT idSucursal
INTO vSuc
FROM sucursal
WHERE nomSuc=vNomSuc;
IF (vSuc IS NOT NULL) THEN
INSERT INTO consultorio(idSucursal,idCons)
VALUES(vSuc,vConsultorio);
COMMIT;
DBMS_OUTPUT.PUT_LINE('Consultorio Creado');
END IF;
EXCEPTION WHEN NO_DATA_FOUND THEN
DBMS_OUTPUT.PUT_LINE('No existe esta sucursal');
RAISE_APPLICATION_ERROR(-20002,'No existe esta sucursal');
END spAddConsultorio;
/

```

```

EXEC spAddConsultorio('Clinica Dental Reluciente', 'H9');
EXEC spAddConsultorio('Odontología Vanguardista', 'J1');
EXEC spAddConsultorio('Clínica Dental Armonía Bucal', 'B2');
EXEC spAddConsultorio('Centro Odontológico Sonrisa Divina', 'C3');
EXEC spAddConsultorio('Consultorio Dental Boca Saludable', 'D4');
EXEC spAddConsultorio('Odontología Integral Dientes Felices', 'E5');
EXEC spAddConsultorio('Clínica Dental Brillo Dental', 'F6');

```

```
EXEC spAddConsultorio('Centro Odontológico Sonrisa Armoniosa', 'G7');
EXEC spAddConsultorio('Consultorio Dental Dientes Sanos y Bellos', 'H8');
EXEC spAddConsultorio('Odontología Avanzada Salud Dental', 'I9');
EXEC spAddConsultorio('Clínica Dental Bucal Integral', 'J5');
```

IDSUCURSAL	ID
12	J1
13	B2
14	C3
15	D4
16	E5
17	F6
18	G7
19	H8
20	I9
21	J5

EQUIPAMIENTO

```
CREATE OR REPLACE PROCEDURE spAddEquipo(
vnumSerie IN VARCHAR2,
vnombreEq IN VARCHAR2,
vnomSuc IN VARCHAR2,
vidCons IN VARCHAR2)
AS
vSuc equipamiento.idSucursal%TYPE;
vCons equipamiento.idCons%TYPE;
vnum equipamiento.numSerie%TYPE;
BEGIN
SELECT idSucursal
INTO vSuc
FROM sucursal
WHERE nomSuc=vnomSuc;
SELECT idCons
INTO vCons
FROM consultorio
WHERE idSucursal=vSuc AND idCons=vidCons;
IF (vSuc IS NOT NULL) THEN
IF (vCons IS NOT NULL) THEN
INSERT INTO equipamiento(numSerie,nombreEq,idSucursal,idCons)
VALUES(vnumSerie,vnombreEq,vSuc,vCons);
COMMIT;
DBMS_OUTPUT.PUT_LINE('Equipamiento agregado' || vNomSuc);
END IF;
END IF;
EXCEPTION
WHEN DUP_VAL_ON_INDEX THEN
DBMS_OUTPUT.PUT_LINE('Error: el Numero de serie ya existe');
RAISE_APPLICATION_ERROR(-20002,'El numero de serie ya existe');
WHEN NO_DATA_FOUND THEN
DBMS_OUTPUT.PUT_LINE('No existe el consultorio en esa sucursal');
RAISE_APPLICATION_ERROR(-20002,'No existe el consultorio en esa sucursal');
WHEN OTHERS THEN
DBMS_OUTPUT.PUT_LINE('Error: ' || SQLERRM);
END spAddEquipo;
```


/

```
EXEC spAddEquipo('ICO2024INTRAORLC01', 'Intraoral Camera Omega', 'Odontología Vanguardista',
'J1');
EXEC spAddEquipo('DCC2024CHAIRDLXC02', 'Dental Chair Comfort', 'Clínica Dental Armonía Bucal',
'B2');
EXEC spAddEquipo('DCM2024CURINGMCH03', 'Dental Curing Machine', 'Centro Odontológico
Sonrisa Divina', 'C3');
EXEC spAddEquipo('DSP2024SCALERPLS04', 'Dental Scaler Plus', 'Consultorio Dental Boca Saludable',
'D4');
EXEC spAddEquipo('DXR2024XRAYRDDNT05', 'X-Ray Radiant Dental', 'Odontología Integral Dientes
Felices', 'E5');
EXEC spAddEquipo('HSH2024HSPDHPXTR06', 'High-Speed Handpiece Extra', 'Clínica Dental Brillo
Dental', 'F6');
EXEC spAddEquipo('DAC2024AUTOCLVCS07', 'Dental Autoclave Classic', 'Centro Odontológico Sonrisa
Armoniosa', 'G7');
EXEC spAddEquipo('APL2024APEXLCTRD08', 'Apex Locator Deluxe', 'Consultorio Dental Dientes Sanos
y Bellos', 'H8');
EXEC spAddEquipo('LCL2024CURINGLGHT09', 'LED Curing Light Hyper', 'Odontología Avanzada Salud
Dental', 'I9');
EXEC spAddEquipo('DSU2024SUCTNUNTP10', 'Dental Suction Unit Pro', 'Clínica Dental Bucal Integral',
'J5');
```

NUMSERIE	NOMBREEQ	FECHAMAN	IDSUCURSAL	ID
DCC2024CHAIRDLXC02	Dental Chair Comfort		13	B2
DCM2024CURINGMCH03	Dental Curing Machine		14	C3
DSP2024SCALERPLS04	Dental Scaler Plus		15	D4
DXR2024XRAYRDDNT05	X-Ray Radiant Dental		16	E5
HSH2024HSPDHPXTR06	High-Speed Handpiece Extra		17	F6
DAC2024AUTOCLVCS07	Dental Autoclave Classic		18	G7
APL2024APEXLCTRD08	Apex Locator Deluxe		19	H8
LCL2024CURINGLGHT09	LED Curing Light Hyper		20	I9
DSU2024SUCTNUNTP10	Dental Suction Unit Pro		21	J5

ODONTOLOGO (Incluyendo horario)

```
CREATE OR REPLACE PROCEDURE agregarodontologo(
vcedulao CHAR,
vnombreo VARCHAR2,
vavpato VARCHAR2,
vavmato VARCHAR2,
vid_sucursal NUMBER,
vhorario VARCHAR2,
vespecialidadP VARCHAR2
)
IS
BEGIN
INSERT INTO odontologo (cedulaO, nombreO, apPatO, apMatO, especialidadP)
VALUES (vcedulao, vnombreo, vavpato, vavmato, vespecialidadP);

INSERT INTO odontoSuc (idSucursal, cedulaO)
VALUES (vid_sucursal, vcedulao);

INSERT INTO horario (idSucursal, cedulaO, horario)
VALUES (vid_sucursal, vcedulao, vhorario);

DBMS_OUTPUT.PUT_LINE('Odontólogo ' || vnombreo || ' ' || vavpato || ' ' || vavmato || ' agregado
```

```
correctamente.');
```

```
EXCEPTION
```

```
WHEN OTHERS THEN
```

```
DBMS_OUTPUT.PUT_LINE('Error: ');
```

```
END;
```

```
/
```

```
EXEC agregar_odontologo('987654', 'María', 'Rodríguez', 'López', 21, 'Lunes a Viernes 9:00 - 17:00',
'Endodoncia');
```

```
EXEC agregar_odontologo('234567', 'Carlos', 'Sánchez', 'Ramírez', 22, 'Martes a Sábado 10:00 -
18:00', 'Periodoncia');
```

```
EXEC agregar_odontologo('345678', 'Ana', 'Gutiérrez', 'Torres', 23, 'Lunes a Viernes 8:00 - 16:00',
'Odontopediatría');
```

```
EXEC agregar_odontologo('456789', 'Luis', 'Hernández', 'Flores', 24, 'Miércoles a Domingo 9:00 -
17:00', 'Cirugía Oral');
```

```
EXEC agregar_odontologo('567890', 'Laura', 'Martínez', 'Gómez', 25, 'Lunes a Viernes 10:00 - 18:00',
'Ortodoncia');
```

```
EXEC agregar_odontologo('678901', 'Pedro', 'Jiménez', 'Díaz', 26, 'Martes a Sábado 8:00 - 16:00',
'Implantología');
```

```
EXEC agregar_odontologo('789012', 'Sofía', 'García', 'Mendoza', 27, 'Lunes a Viernes 9:00 - 17:00',
'Rehabilitación Oral');
```

```
EXEC agregar_odontologo('890123', 'Diego', 'Ruiz', 'Vargas', 28, 'Miércoles a Domingo 10:00 - 18:00',
'Odontología Estética');
```

```
EXEC agregar_odontologo('901234', 'Marta', 'Moreno', 'Castillo', 29, 'Lunes a Viernes 8:00 - 16:00',
'Odontología Preventiva');
```

```
EXEC agregar_odontologo('012345', 'Javier', 'Navarro', 'Romero', 30, 'Martes a Sábado 9:00 - 17:00',
'Odontología Geriátrica');
```

CEDULA NOMBREO ESPECIALIDADP	APPATO	APMATO
-----	-----	-----
234567 Carlos Periodoncia	Sánchez	Ramírez
345678 Ana Odontopediatría	Gutiérrez	Torres
456789 Luis Cirugía Oral	Hernández	Flores
567890 Laura Ortodoncia	Martínez	Gómez

TRATAMIENTO

```
CREATE OR REPLACE PROCEDURE agregar_tratamiento(
vabreviatura VARCHAR2,
vnombre_trat VARCHAR2,
vcosto NUMBER)
IS
BEGIN
INSERT INTO tratamiento (abreviatura, nombreTrat, costo)
VALUES (vabreviatura, vnombre_trat, vcosto);
DBMS_OUTPUT.PUT_LINE('Tratamiento agregado correctamente.');
```

```
EXCEPTION
```

```
WHEN DUP_VAL_ON_INDEX THEN
```

```
DBMS_OUTPUT.PUT_LINE('Error: el tratamiento ya existe.');
```

```
WHEN OTHERS THEN
```

```
DBMS_OUTPUT.PUT_LINE('Error: ');
```

```
END;
```

/

```
EXEC agregar_tratamiento('CPR', 'Coronas de Porcelana', 850);
EXEC agregar_tratamiento('EDS', 'Endodoncia Simple', 450);
EXEC agregar_tratamiento('EDC', 'Endodoncia Compleja', 900);
EXEC agregar_tratamiento('FRP', 'Férula de Relajación', 350);
EXEC agregar_tratamiento('APD', 'Apicectomía', 1200);
EXEC agregar_tratamiento('RRC', 'Reconstrucción de Reborde', 750);
EXEC agregar_tratamiento('IND', 'Injerto de Hueso', 1500);
EXEC agregar_tratamiento('EXR', 'Extracción de Raíz', 250);
EXEC agregar_tratamiento('TDS', 'Tratamiento de Sensibilidad', 180);
EXEC agregar_tratamiento('VTR', 'Vestibuloplastia', 600);
```

IDTRATAMIENTO	ABR	NOMBRETRAT	COSTO
12	EDS	Endodoncia Simple	450
13	EDC	Endodoncia Compleja	900
14	FRP	Férula de Relajación	350
15	APD	Apicectomía	1200
16	RRC	Reconstrucción de Reborde	750
17	IND	Injerto de Hueso	1500
18	EXR	Extracción de Raíz	250
19	TDS	Tratamiento de Sensibilidad	180
20	VTR	Vestibuloplastia	600

REGISTRO (Incluyendo el tratamiento)

```
CREATE OR REPLACE PROCEDURE spAddRegistroHist(
vCedulaO IN VARCHAR2,
vNomSuc IN VARCHAR2,
vidCons IN VARCHAR2,
vcedulaAF IN VARCHAR2,
vfechaC IN VARCHAR2,
vTratamiento IN VARCHAR2,
vPago IN NUMERIC,
vFechaIn IN VARCHAR2,
vFin IN VARCHAR2)
AS
vSuc registro.idSucursal%TYPE;
vCons registro.idCons%TYPE;
vidT historia_med.idTratamiento%TYPE;
vConsHist historia_med.idConsulta%TYPE;
vNS registro.idSucursal%TYPE;
vO registro.cedulaO%TYPE;
vt historia_med.idTratamiento%TYPE;
vOT registro.cedulaO%TYPE;
BEGIN
SELECT idTratamiento
INTO vidT
FROM tratamiento
WHERE nombreTrat=vTratamiento;
SELECT idTratamiento, CedulaO
INTO vT,vOT
FROM odontoTrab
```

```

WHERE idTratamiento=vidT AND cedulaO=vCedulaO;
SELECT idSucursal
INTO vSuc
FROM sucursal
WHERE nomSuc=vNomSuc;
SELECT idSucursal,cedulaO
INTO vNS,vO
FROM odontoSuc
WHERE idSucursal=vSuc AND cedulaO=vcedulaO;
SELECT idCons
INTO vCons
FROM consultorio
WHERE idSucursal=vSuc AND idCons=vidCons;
IF (vNS IS NOT NULL) THEN
IF (vCons IS NOT NULL) THEN
IF (vO IS NOT NULL) THEN
INSERT INTO registro(cedulaO,idSucursal,idCons,cedulaAF,fechaConsulta)
VALUES(vO,vNS,vCons,vcedulaAF,vfechaC);
COMMIT;
DBMS_OUTPUT.PUT_LINE('Consulta agendada');
END IF;
END IF;
END IF;
SELECT idConsulta
INTO vConsHist
FROM registro
WHERE cedulaO=vO AND idSucursal=vNS AND idCons=vCons
AND cedulaAF=vcedulaAF AND fechaConsulta=vFechaC;
IF (vOT IS NOT NULL) THEN
IF (vConsHist IS NOT NULL) THEN
INSERT INTO historia_Med (cedulaAF, idTratamiento, cedulaO, pago, fechaInicioT, finalizado,
idConsulta)
VALUES (vcedulaAF,vT, vOT, ftPagoTD(vcedulaAF,vT), vFechaIn,vFin, vConsHist);
COMMIT;
DBMS_OUTPUT.PUT_LINE('Tratamiento agendado');
END IF;
END IF;
EXCEPTION
WHEN DUP_VAL_ON_INDEX THEN
DBMS_OUTPUT.PUT_LINE('Error: Esta consulta ya esta agendada ');
WHEN NO_DATA_FOUND THEN
DBMS_OUTPUT.PUT_LINE('No existe el consultorio en esa sucursal o el odontologo no trabaja en la
sucursal');
RAISE_APPLICATION_ERROR(-20003,'No existe el consultorio o el odontologo no trabaja ahi');
WHEN OTHERS THEN
DBMS_OUTPUT.PUT_LINE('Datos no validos');
RAISE_APPLICATION_ERROR(-20003,'Datos no validos');
END spAddRegistroHist;
/

```

AFILIADO

```

CREATE OR REPLACE PROCEDURE agregar_afiliado(
vcedulaAF IN CHAR,
vnombre IN VARCHAR2,
vap_paterno IN VARCHAR2,
vap_materno IN VARCHAR2,
vcuota IN NUMBER,
vtipo_afiliado IN CHAR,
vnum_convenio IN CHAR,
vrebaja IN NUMBER,
vtelefono IN CHAR)
AS
vAfil afiliado.cedulaAF%TYPE;
BEGIN
INSERT INTO afiliado (cedulaAF, nombreAF, apPatAF, apMatAF, cuota, tipoAF)
VALUES (vcedulaAF, vnombre, vap_paterno, vap_materno, vcuota, vtipo_afiliado);
IF vtipo_afiliado = 'G' THEN
INSERT INTO grupal (cedulaAF, numConvenio, rebaja)
VALUES (vcedulaAF, vnum_convenio, vrebaja);
END IF;
INSERT INTO telefono (cedulaAF, telefono)
VALUES (vcedulaAF, vtelefono);
COMMIT;
EXCEPTION
WHEN OTHERS THEN
RAISE_APPLICATION_ERROR(-20001, 'Error al agregar el afiliado: ' || vnombre);
DBMS_OUTPUT.PUT_LINE('Afiliado Creado' || vnombre);
END agregar_afiliado;
/

EXEC agregar_afiliado('TORJ8708', 'Javier', 'Torres', 'Rodríguez', 2000, 'I', NULL, NULL, '5512345678');
EXEC agregar_afiliado('PERA9009', 'Andrea', 'Pérez', 'Aguilar', 2000, 'I', NULL, NULL, '5567890123');
EXEC agregar_afiliado('CASTD0110', 'Diego', 'Castillo', 'Torres', 2000, 'I', NULL, NULL, '5523456789');
EXEC agregar_afiliado('GUZL8011', 'Laura', 'Guzmán', 'López', 2000, 'I', NULL, NULL, '5598765432');
EXEC agregar_afiliado('HERV7112', 'Víctor', 'Hernández', 'Vázquez', 2000, 'I', NULL, NULL,
'5512349876');
EXEC agregar_afiliado('RAMF6213', 'Fernanda', 'Ramírez', 'Flores', 2000, 'G', 'CON00004', 15,
'5567891234');
EXEC agregar_afiliado('SANM5314', 'Manuel', 'Sánchez', 'Mora', 2000, 'G', 'CON00004', 15,
'5523457890');
EXEC agregar_afiliado('MART4415', 'Tania', 'Martínez', 'Torres', 2000, 'G', 'CON00004', 15,
'5598760123');
EXEC agregar_afiliado('CRUZ3516', 'Javier', 'Cruz', 'Zavala', 2000, 'G', 'CON00005', 10, '5512345609');
EXEC agregar_afiliado('RODR2617', 'Rodrigo', 'Rodríguez', 'Díaz', 2000, 'G', 'CON00005', 10,
'5567898765');

```

CEDULA AF	NOMBRE AF	APPATAF	APMATAF
CUOTA	T		
-----	-----	-----	-----
CRUZ3516	Javier	Cruz	Zavala
2000	G		
RODR2617	Rodrigo	Rodríguez	Díaz
2000	G		

TELEFONOS (Agrega los telefonos a los afiliados)

```

CREATE OR REPLACE PROCEDURE agregar_telefono(
vcedulaAf IN CHAR,
vtelefono IN CHAR)
AS
vtel NUMBER;
BEGIN
SELECT COUNT(*) INTO vtel
FROM afiliado
WHERE cedulaAF = vcedulaAf;
IF vtel = 0 THEN
RAISE_APPLICATION_ERROR(-20001, 'El afiliado con cédula no existe.');
```

```

END IF;
INSERT INTO telefono (cedulaAF, telefono)
VALUES (vcedulaAf, vtelefono);
COMMIT;
EXCEPTION
WHEN OTHERS THEN
RAISE_APPLICATION_ERROR(-20001, 'Error ');
END agregar_telefono;
/

EXEC agregar_telefono('TORJ8708', '5512345679');
EXEC agregar_telefono('TORJ8708', '5512345670');
EXEC agregar_telefono('PERA9009', '5567890124');
EXEC agregar_telefono('PERA9009', '5567890125');
EXEC agregar_telefono('CASTD0110', '5523456780');
EXEC agregar_telefono('CASTD0110', '5523456781');
EXEC agregar_telefono('GUZL8011', '5598765433');
EXEC agregar_telefono('GUZL8011', '5598765434');
EXEC agregar_telefono('HERV7112', '5512349877');
EXEC agregar_telefono('HERV7112', '5512349878');
EXEC agregar_telefono('RAMF6213', '5567891235');
EXEC agregar_telefono('SANM5314', '5523457891');
EXEC agregar_telefono('MART4415', '5598760124');
EXEC agregar_telefono('MART4415', '5598760125');
EXEC agregar_telefono('CRUZ3516', '5512345610');
EXEC agregar_telefono('CRUZ3516', '5512345611');
EXEC agregar_telefono('RODR2617', '5567898766');
EXEC agregar_telefono('RODR2617', '5567898767');
```

CEDULA AF	TELEFONO
RIVG8102	5598761234
RODH8203	5598765123
RODR2617	5567898765
RODR2617	5567898766
RODR2617	5567898767
SACM7707	5587659876
SACM7707	5598761234
SANM5314	5523457890
SANM5314	5523457891
TORJ8708	5512345670
TORJ8708	5512345678

SUCURSALES (Agrega sucursales de trabajo a los Odontologos)

```
CREATE OR REPLACE PROCEDURE spAddSO(
vCedulaO IN VARCHAR2,
vNomSuc IN VARCHAR2)
AS
vIdSuc sucursal.idSucursal%TYPE;
BEGIN
SELECT idSucursal
INTO vIdSuc
FROM sucursal
WHERE nomSuc=vNomSuc;

INSERT INTO odontoSuc
VALUES(vIdSuc,vCedulaO);
EXCEPTION
WHEN NO_DATA_FOUND THEN
DBMS_OUTPUT.PUT_LINE('La sucursal o el odontologo no exixsten');
WHEN DUP_VAL_ON_INDEX THEN
DBMS_OUTPUT.PUT_LINE('Error: EL odontolo ya trabaja en esta sucursal');
END spAddSO;
/
```

TRATAMIENTOS (Agrega tratamientos capaz de realizar a los Oodntologos)

```
CREATE OR REPLACE PROCEDURE spAddOT(
vCedulaO IN VARCHAR2,
vnomTrat VARCHAR2)
AS
vidTrat tratamiento.idTratamiento%TYPE;
BEGIN
SELECT idTratamiento
INTO vidTrat
FROM tratamiento
WHERE nombreTrat=vnomTrat;
INSERT INTO odontoTrab
VALUES(vidTrat,vCedulaO);
EXCEPTION
WHEN NO_DATA_FOUND THEN
DBMS_OUTPUT.PUT_LINE('El tartamiento o el odontologo no exixsten');
WHEN DUP_VAL_ON_INDEX THEN
DBMS_OUTPUT.PUT_LINE('Error: EL odontolo ya realiza este tratamiento');
END spAddOT;
/
```

II. Actualizaciones a través de Procedimientos

EQUIPAMIENTO (Actualiza la fecha de mantenimiento del equipo de forma individual)

```
CREATE OR REPLACE PROCEDURE spMantenimientoEQ(
```



```

vNumSerie IN VARCHAR2,
vFechaN IN DATE)
AS
vexist NUMBER;
BEGIN
SELECT COUNT(*) INTO vexist
FROM equipamiento
WHERE numSerie = vNumSerie;
IF vexist = 0 THEN
RAISE_APPLICATION_ERROR(-20001, 'El equipo no existe.');
```

```

END IF;
UPDATE equipamiento
SET fechaMantenimiento = vFechaN
WHERE numSerie = vNumSerie;
COMMIT;
EXCEPTION
WHEN OTHERS THEN
RAISE_APPLICATION_ERROR(-20002, 'Error');
```

```

END spMantenimientoEQ;
/
```

NUMSERIE	NOMBREEQ	FECHAMAN	IDSUCURSAL	ID
USP2024SCALERPR001	Ultrasonic Scaler Pro	15/03/24	10	D7
DXM2024XRAYMACH02	Digital X-Ray Machine	22/04/24	1	A1
HS2024HSPDHPCE03	High-Speed Handpiece	10/05/24	6	F2

Actualizado:

NUMSERIE	NOMBREEQ	FECHAMAN	IDSUCURSAL	ID
USP2024SCALERPR001	Ultrasonic Scaler Pro	15/05/24	10	D7
DXM2024XRAYMACH02	Digital X-Ray Machine	22/04/24	1	A1

EQUIPAMIENTO (Actualiza la fecha de mantenimiento del equipamiento por consultorio)

```

CREATE OR REPLACE PROCEDURE spMantenimientoSC(
vNomSuc VARCHAR2,
vNumCons VARCHAR2,
vFechaN DATE)
AS
vNS sucursal.idSucursal%TYPE;
vNumSerie equipamiento.numSerie%TYPE;
BEGIN
SELECT idSucursal
INTO vNS
FROM sucursal
WHERE nomSuc=vNomSuc;
SELECT numSerie
INTO vNumSerie
FROM equipamiento
WHERE idSucursal=vNS AND idCons=vNumCons;
IF vNumSerie IS NOT NULL THEN
UPDATE equipamiento
SET fechaMantenimiento=vFechaN
WHERE numSerie=vNumSerie;
```

```

END IF;
EXCEPTION WHEN NO_DATA_FOUND THEN
DBMS_OUTPUT.PUT_LINE('Este equipo no se encuentra en el consultorio de la sucursal');
END spMantenimientoSC;
/

```

```

EXEC spMantenimientoSC('Clínica Dental Sonrisa Perfecta', 'A1', TO_DATE('01/06/2024',
'DD/MM/YYYY'));

```

AFILIADOS (Aumenta la cuota para todos los afiliados)

```

CREATE OR REPLACE PROCEDURE spCuotaUP(
vCuotaN IN NUMERIC)
AS
BEGIN
UPDATE afiliado
SET cuota=vCuotaN;
END spCuotaUP;
/

```

HISTORIA_MED (Cambia es status del tratamiento a finalizado)

```

CREATE OR REPLACE PROCEDURE spTratamientoFin(
vCedulaAF VARCHAR2,
vCedulaO VARCHAR2,
vnomTrat VARCHAR2)
AS
vidTrat tratamiento.idTratamiento%TYPE;
BEGIN
SELECT idTratamiento
INTO vidTrat
FROM tratamiento
WHERE nombreTrat=vnomTrat;
UPDATE historia_Med
SET finalizado='S'
WHERE cedulaO=vCedulaO AND cedulaAF=vCedulaAF AND idTratamiento=vidTrat;
EXCEPTION WHEN NO_DATA_FOUND THEN
DBMS_OUTPUT.PUT_LINE('Este tratamiento realizado por este doctor para este paciente no existe');
END spTratamientoFin;
/

```

III. Borrar

SUCURSAL

```

CREATE OR REPLACE PROCEDURE spSucursalD(
vnomSuc VARCHAR2)
AS
vIdSuc sucursal.idSucursal%TYPE;
BEGIN
DELETE FROM sucursal
WHERE nomSuc = vNomSuc;
END spSucursalD;

```

/

H. Actividades

```
SET SERVEROUTPUT ON
SET SERVEROUTPUT ON SIZE UNLIMITED
```

1. Ejercicio 1

Al realizarse una consulta se debe registrar en la historia clínica del paciente los datos correspondientes, así como el tratamiento a realizarse.

Calcula el pago de del tratamiento incluyendo el descuento (afiliados grupales)

```
CREATE OR REPLACE FUNCTION ftPagoTD(
vcedulaAF IN VARCHAR2,
vidTrat IN NUMBER)
RETURN NUMBER
IS
vCosto tratamiento.costo%TYPE;
vtipoAF CHAR(1);
vDesc NUMERIC(2,0);
vMontoD NUMBER(10,2);
vTotal NUMBER(10,2);
BEGIN
SELECT costo
INTO vCosto
FROM tratamiento
WHERE idTratamiento=vidTrat;

SELECT tipoAF
INTO vtipoAF
FROM afiliado
WHERE cedulaAF=vcedulaAF;
IF vtipoAF='G' THEN
SELECT rebaja
INTO vDesc
FROM grupal
WHERE cedulaAF=vcedulaAF;
ELSE
vDesc := 0;
END IF;
vMontoD:=vCosto*(vDesc/100);
vTotal:=vCosto-vMontoD;
RETURN(vTotal);
END ftPagoTD;
/
```

Realiza el registro de la consulta y agenda el tratamiento

```
CREATE OR REPLACE PROCEDURE spAddRegistroHist(
vCedulaO IN VARCHAR2,
```

```

vNomSuc IN VARCHAR2,
vidCons IN VARCHAR2,
vcedulaAF IN VARCHAR2,
vfechaC IN VARCHAR2,
vTratamiento IN VARCHAR2,
vPago IN NUMERIC,
vFechaIn IN VARCHAR2,
vFin IN VARCHAR2)
AS
vSuc registro.idSucursal%TYPE;
vCons registro.idCons%TYPE;
vidT historia_med.idTratamiento%TYPE;
vConsHist historia_med.idConsulta%TYPE;
vNS registro.idSucursal%TYPE;
vO registro.cedulaO%TYPE;
vt historia_med.idTratamiento%TYPE;
vOT registro.cedulaO%TYPE;
BEGIN
SELECT idTratamiento
INTO vidT
FROM tratamiento
WHERE nombreTrat=vTratamiento;
SELECT idTratamiento, CedulaO
INTO vT,vOT
FROM odontoTrab
WHERE idTratamiento=vidT AND cedulaO=vCedulaO;
SELECT idSucursal
INTO vSuc
FROM sucursal
WHERE nomSuc=vNomSuc;
SELECT idSucursal,cedulaO
INTO vNS,vO
FROM odontoSuc
WHERE idSucursal=vSuc AND cedulaO=vcedulaO;
SELECT idCons
INTO vCons
FROM consultorio
WHERE idSucursal=vSuc AND idCons=vidCons;
IF (vNS IS NOT NULL) THEN
IF (vCons IS NOT NULL) THEN
IF (vO IS NOT NULL) THEN
INSERT INTO registro(cedulaO,idSucursal,idCons,cedulaAF,fechaConsulta)
VALUES(vO,vNS,vCons,vcedulaAF,vfechaC);
COMMIT;
DBMS_OUTPUT.PUT_LINE('Consulta agendada');
END IF;
END IF;
END IF;
SELECT idConsulta
INTO vConsHist
FROM registro

```

```

WHERE cedulaO=vO AND idSucursal=vNS AND idCons=vCons
AND cedulaAF=vcedulaAF AND fechaConsulta=vFechaC;
IF (vOT IS NOT NULL) THEN
IF (vConsHist IS NOT NULL) THEN
INSERT INTO historia_Med (cedulaAF, idTratamiento, cedulaO, pago, fechaInicioT, finalizado,
idConsulta)
VALUES (vcedulaAF,vT, vOT, ftPagoTD(vcedulaAF,vT), vFechaIn,vFin, vConsHist);
COMMIT;
DBMS_OUTPUT.PUT_LINE('Tratamiento agendado');
END IF;
END IF;
EXCEPTION
WHEN DUP_VAL_ON_INDEX THEN
DBMS_OUTPUT.PUT_LINE('Error: Esta consulta ya esta agendada ');
WHEN NO_DATA_FOUND THEN
DBMS_OUTPUT.PUT_LINE('No existe el consultorio en esa sucursal o el odontologo no trabaja en la
sucursal');
RAISE_APPLICATION_ERROR(-20003,'No existe el consultorio o el odontologo no trabaja ahi');
WHEN OTHERS THEN
DBMS_OUTPUT.PUT_LINE('Datos no validos');
RAISE_APPLICATION_ERROR(-20003,'Datos no validos');
END spAddRegistroHist;
/

```

Por ejemplo:

```

EXEC spAddRegistroHist('778899','Centro Odontológico
Dentisano','G7','JIMR7910','27-05-24','Cirugía Ortognática',2630,'30-05-24','N');

```

```
SQL> select * from vwhistorial;
```

AFILIADO	ODONTO	SUCURSAL	CO	FECHA	CON	TRATAMIENTO	PAGO	FECHA_IN	F
RODH8203	654321	Centro Odontológico Diente de Oro	D1	08/12/08		Implante Dental	10000	08/12/08	S
MARM8506	654321	Centro Odontológico Diente de Oro	D1	14/06/06		Implante Dental	8000	14/06/06	S
MORR8608	654321	Odontologia Avanzada Blanca	C1	02/10/20		Implante Dental	6000	02/10/20	S
MORR8608	998877	Centro Odontológico Dentisano	87	02/10/20		Implante Dental	12000	02/10/20	S
GONJ8501	112233	Centro Odontológico Dentisano	87	22/11/22		Extracción de Dientes	2000	22/11/22	S
RIVG8102	112233	Centro Odontológico Dentisano	G7	11/11/21		Periodoncia	400	11/11/21	S
GONJ8501	889900	Clinica Dental Sonrisa Perfecta	A1	12/05/24		Cirugía Ortognática	1000	12/05/24	N
HERS8903	889900	Clinica Dental Sonrisa Perfecta	A1	14/07/21		Cirugía Ortognática	30000	14/07/21	N
ALVM8001	778899	Odontologia Avanzada Blanca	C1	07/03/21		Cirugía Ortognática	20000	07/03/21	N
CALR8405	778899	Odontologia Avanzada Blanca	C1	03/06/03		Cirugía Ortognática	28000	03/06/03	S
JIMR7910	778899	Centro Odontológico Dentisano	G7	27/05/24		Cirugía Ortognática	22500	30/05/24	N

2. Ejercicio 2

Se desea conocer el monto total de los pagos realizados de cada uno de los afiliados, obtenido en forma mensual.

Realiza la suma del total de tratamientos y la cuota para un afiliado

```

CREATE OR REPLACE FUNCTION ftMontoTotalAF(
vcedulaAF IN VARCHAR2)
RETURN NUMBER
IS
vTotal NUMBER(11,2);
vCuota NUMERIC(10,2);
vMonto NUMBER(11,2);
vCuenta NUMBER(5):=0;
BEGIN

```

```

SELECT cuota
INTO vCuota
FROM afiliado
WHERE cedulaAF=vcedulaAF;
SELECT COUNT(cedulaAF)
INTO vCuenta
FROM historia_Med
WHERE cedulaAF=vcedulaAF;
IF vCuenta>0 THEN
SELECT SUM(pago)
INTO vMonto
FROM historia_Med
GROUP BY cedulaAF
HAVING cedulaAF=vcedulaAF;
ELSE
vMonto:=0;
END IF;
vTotal:=vCuota + vMonto;
RETURN(vTotal);
END ftMontoTotalAF;
/

```

Realiza y muestra la suma total para cada afiliado

```

CREATE OR REPLACE PROCEDURE totalAF
AS
vNomAF VARCHAR2(50);
vApPat VARCHAR2(50);
vApMat VARCHAR2(50);
vcedulaAF afiliado.cedulaAF%TYPE;
vTotal NUMBER(11,2);
CURSOR regAF
IS
SELECT *
FROM afiliado;
rAF afiliado%ROWTYPE;
BEGIN
FOR rAF IN regAF LOOP
DBMS_OUTPUT.PUT_LINE(rAF.cedulaAF);
vTotal:=ftMontoTotalAF(rAF.cedulaAF);
DBMS_OUTPUT.PUT_LINE(rAF.nombreAF || ' ' || rAF.apPatAF || ' ' || rAF.apMatAF || ' ' || 'Total
pagado: ' || vTotal);
END LOOP;
END totalAF;
/

```

Por ejemplo:

```
EXEC totalAF;
```

```
SQL> EXEC totalAF;
GOMJ8501
Juan Gómez Martínez      Total pagado: 2500
LOPF7902
Fernanda López Pérez     Total pagado: 2500
HERS8903
Sofía Hernández Ruiz    Total pagado: 2500
GARR7405
Roberto García Ramírez   Total pagado: 2500
TORR8206
Raquel Torres Rojas      Total pagado: 2500
SACH7707
Miguel Sánchez Castillo  Total pagado: 2500
MORR8608
Ramón Moreno Romero      Total pagado: 2500
JIMR7910
Rebeca Jiménez Martínez  Total pagado: 25000
ALVM8001
Víctor Álvarez Muñoz     Total pagado: 2500
RIVG8102
Gabriela Rivera Guzmán   Total pagado: 2500
RODH8203
Héctor Rodríguez Hernández Total pagado: 2500
MONP8304
Patricia Montes Pérez    Total pagado: 2500
CALR8405
Ricardo Calderón Lara    Total pagado: 2500
MARM8506
Mónica Martínez Mendoza Total pagado: 2500
VALC8607
Carlos Valdés Cárdenas   Total pagado: 2500

Procedimiento PL/SQL terminado correctamente.
```

3. Ejercicio 3

Muestre el número del convenio de afiliación y el porcentaje de rebaja que se aplica a la cuota mensual.

```
CREATE OR REPLACE VIEW convenios AS
SELECT DISTINCT numConvenio, rebaja || '%' as descuento
FROM grupal;
```

Por ejemplo:

```
SQL> select * from convenios;

NUMCONVE  DESCUENTO
-----
CON000001 25%
CON000002 15%
CON000003 10%
```

4. Ejercicio 4

Muestre los datos de las sucursales y los tratamientos que se realizan.

```
CREATE OR REPLACE VIEW sucursaltrat AS
SELECT s.idSucursal, s.nomSuc, s.ciudad, s.calle, s.numero,
LISTAGG(t.nombreTrat, ', ') WITHIN GROUP (ORDER BY t.nombreTrat) AS tratamiento
FROM sucursal s
JOIN odontoSuc os ON s.idSucursal = os.idSucursal
JOIN odontoTrab ot ON os.cedulaO = ot.cedulaO
JOIN tratamiento t ON ot.idTratamiento = t.idTratamiento
GROUP BY s.idSucursal, s.nomSuc, s.ciudad, s.calle, s.numero
ORDER BY s.idSucursal;
```


Por ejemplo:

IDSUCURSAL NUMERO	NOMSUC	CIUDAD	CALLE
TARTAMIENTO			
45	1 Clínica Dental Sonrisa Perfecta	Madrid	Calle Mayor
Cirugía Ortognática			
67	2 Centro Odontológico Diente de Oro	Valencia	Calle de la Paz
Carillas de Porcelana, Implante Dental			
233	3 Consultorio Dental Brillante	Barcelona	Avenida Diagonal
Cirugía Ortognática, Extracción de Dientes, Implante Dental			
IDSUCURSAL NUMERO	NOMSUC	CIUDAD	CALLE
TARTAMIENTO			
102	4 Odontología Avanzada Blanca	Sevilla	Calle Sierpes
Carillas de Porcelana, Cirugía Ortognática, Cirugía Ortognática, Implante Dental			
995	8 Centro Odontológico Dentisano	Santiago	Calle Huérfanos
Cirugía Ortognática, Extracción de Dientes, Extracción de Dientes, Implante Dental, Periodoncia			

5. Ejercicio 5

Realizar una función que realice el cálculo del costo del tratamiento de los afiliados grupales.

SET SERVEROUTPUT ON SIZE UNLIMITED (necesario para ver la salida)

```
CREATE OR REPLACE FUNCTION costo_grupal(
p_cedula_af CHAR,
p_id_tratamiento NUMBER)
RETURN NUMBER
IS
v_costo_tratamiento NUMBER;
v_rebaja NUMBER;
BEGIN
-- Obtener el costo del tratamiento
SELECT costo
INTO v_costo_tratamiento
FROM tratamiento
WHERE idTratamiento = p_id_tratamiento;
-- Verificar si el afiliado es grupal
SELECT rebaja
INTO v_rebaja
FROM grupal
WHERE cedulaAF = p_cedula_af;
-- Calcular el costo con descuento
RETURN v_costo_tratamiento - (v_costo_tratamiento * (v_rebaja / 100));
EXCEPTION
WHEN NO_DATA_FOUND THEN
-- Si el afiliado no es grupal, no se aplica descuento
RETURN v_costo_tratamiento;
WHEN OTHERS THEN
RAISE_APPLICATION_ERROR(-20001, 'Error al calcular el costo del tratamiento: ' || SQLERRM);
END;
```

/

Por ejemplo:

```
DECLARE
v_costo NUMBER;
BEGIN
v_costo := costo_grupal('SACM7707', 1); -- Afiliado grupal con rebaja del 25%
DBMS_OUTPUT.PUT_LINE('Costo del tratamiento para el afiliado grupal: ' || v_costo);
END;
/
```

Costo del tratamiento para el afiliado grupal: 750

Procedimiento PL/SQL terminado correctamente.

6. Ejercicio 6

Muestre los horarios de atención de cada uno de los odontólogos por sucursal.

```
CREATE OR REPLACE VIEW sucursalhorario AS
SELECT s.nomSuc, o.cedulaO, o.nombreO, o.apPatO, o.apMatO,
LISTAGG(h.horario, ' | ') WITHIN GROUP (ORDER BY h.horario) AS horarios
FROM sucursal s
JOIN odontoSuc os ON s.idSucursal = os.idSucursal
JOIN odontologo o ON os.cedulaO = o.cedulaO
JOIN horario h ON os.idSucursal = h.idSucursal AND os.cedulaO = h.cedulaO
GROUP BY s.nomSuc, o.cedulaO, o.nombreO, o.apPatO, o.apMatO
ORDER BY s.nomSuc;
```

Por ejemplo:

NOMSUC APMATO	CEDULA NOMBREO	APPATO

HORARIOS		

Centro Odontológico Dentisano Fernández sab de 11:00-17:00	112233 Laura	Gómez
Centro Odontológico Dentisano Díaz lun, vie de 10:00-16:00	998877 Franciso	Morales
Centro Odontológico Dentisano Torres mar,mier de 7:00-14:00	778899 Carmen Elena	Ramírez

NOMSUC	CEDULA NOMBREO	APPATO
APMATO		

HORARIOS		

Centro Odontológico Diente de Oro Ramírez lun, mie, vie de 11:00-15:00	683377 Marisol	Figueroa
Centro Odontológico Diente de Oro Martínez lun, mar, jue de 16:00-20:00 vie,sab de 7:00-14:00	654321 Juan Carlos	Pérez
Centro Odontológico Diente de Oro Ruiz lun, mie, vie de 11:00-15:00	686677 Carolina	Figueroa

7. Ejercicio 7

Muestre los nombres de los afiliados y todos los datos de los tratamientos que les fueron

aplicados por sucursal.

```
CREATE OR REPLACE VIEW afiliadotrat AS
SELECT s.nomSuc, a.nombreAF, a.apPatAF, a.apMatAF, t.*
FROM sucursal s
JOIN registro r ON s.idSucursal = r.idSucursal
LEFT JOIN historia_Med h ON r.idConsulta = h.idConsulta
JOIN afiliado a ON r.cedulaAF = a.cedulaAF
JOIN tratamiento t ON h.idTratamiento = t.idTratamiento
ORDER BY s.nomSuc;
```

Por ejemplo:

```
SELECT * FROM afiliadotrat ;
```

```
SQL> SET LINE 200
SQL> SELECT * FROM afiliadotrat;
```

NOMSUC	NOMBREAF	APPATAF		
APMATAF	IDTRATAMIENTO	ABR	NOMBRETRAT	COSTO
Centro Odontológico Dentisano Martínez	Rebeca	10	CO Cirugía Ortognática	Jiménez 30000

8. Ejercicio 8

Se desea llevar el control por consultorio del mantenimiento que se realiza a su equipamiento, registrando para cada equipo la fecha en que se realizó dicho mantenimiento.

```
ALTER TABLE equipamiento
ENABLE ALL TRIGGERS;
```

```
CREATE TABLE bitacoraMant(
numSerie CHAR(20),
fecha DATE
);
```

```
CREATE OR REPLACE TRIGGER tgMantenimientoEQ
AFTER INSERT OR UPDATE
ON equipamiento
FOR EACH ROW
BEGIN
INSERT INTO bitacoraMant
VALUES(:NEW.numSerie,:NEW.fechaMantenimiento);
END tgMantenimientoEQ;
/
```

Por ejemplo:

```
SELECT * FROM bitacoraMant;
```

```
SQL> SELECT * FROM bitacoraMant;
```

NUMSERIE	FECHA
DXM2024XRAYMACH02	01/06/24

9. Ejercicio 9

```
CREATE OR REPLACE VIEW odontosuct AS
SELECT o.cedulaO, o.nombreO, o.apPatO, o.apMatO, o.especialidadP,
LISTAGG(DISTINCT c1.nombreTrat, ', ') WITHIN GROUP (ORDER BY c1.nombreTrat) AS tratamientos,
LISTAGG(DISTINCT s.nomSuc, ', ') WITHIN GROUP (ORDER BY s.nomSuc) AS sucursales
FROM odontologo o
JOIN odontoSuc os ON o.cedulaO = os.cedulaO
JOIN sucursal s ON os.idSucursal = s.idSucursal
LEFT JOIN (SELECT ot.cedulaO, t.nombreTrat
FROM odontoTrab ot
LEFT JOIN tratamiento t ON ot.idTratamiento = t.idTratamiento
) c1 ON c1.cedulaO=o.cedulaO
GROUP BY o.cedulaO, o.nombreO, o.apPatO, o.apMatO, o.especialidadP;
```

Por ejemplo:

NOMSUC APMATO	CEDULA NOMBREO	APPATO

HORARIOS		

Centro Odontológico Dentisano Fernández sab de 11:00-17:00	112233 Laura	Gómez
Centro Odontológico Dentisano Díaz lun, vie de 10:00-16:00	998877 Francsico	Morales
Centro Odontológico Dentisano Torres mar,mier de 7:00-14:00	778899 Carmen Elena	Ramírez

NOMSUC APMATO	CEDULA NOMBREO	APPATO

HORARIOS		

Centro Odontológico Diente de Oro Ramírez lun, mie, vie de 11:00-15:00	683377 Marisol	Figueroa
Centro Odontológico Diente de Oro Martínez lun, mar, jue de 16:00-20:00 vie,sab de 7:00-14:00	654321 Juan Carlos	Pérez
Centro Odontológico Diente de Oro Ruiz lun, mie, vie de 11:00-15:00	686677 Carolina	Figueroa

NOMSUC APMATO	CEDULA NOMBREO	APPATO

HORARIOS		

Odontología Avanzada Blanca Martínez jue de 15:00-20:00	654321 Juan Carlos	Pérez
Odontología Avanzada Blanca Torres vie, sab, dom de 12:00-17:00	889900 Carmen Elena	Ramirez
Odontología Avanzada Blanca Torres Lun, mie de 8:00-14:00 mar, jue de 8:00-12:00	778899 Carmen Elena	Ramirez

10. Ejercicio 10

Obtenga el monto mensual obtenido de los tratamientos realizados por sucursal

```
CREATE OR REPLACE VIEW vmonto AS
SELECT idSucursal, nomSuc, SUM(costo) AS montoMensual
FROM (SELECT s.idSucursal, s.nomSuc, t.idTratamiento, t.nombreTrat, t.costo
FROM sucursal s
JOIN odontoSuc os ON s.idSucursal = os.idSucursal
JOIN odontoTrab ot ON os.cedulaO = ot.cedulaO
JOIN tratamiento t ON ot.idTratamiento = t.idTratamiento) c1
WHERE TRUNC(SYSDATE, 'MM') = TRUNC(SYSDATE, 'MM')
GROUP BY idSucursal, nomSuc;
```

Por ejemplo:

```
SQL> select * from vmonto;
```

IDSUCURSAL	NOMSUC	MONDOMENSUAL

2	Centro Odontológico Diente de Oro	12900
8	Centro Odontológico Dentisano	47450
4	Odontología Avanzada Blanca	72900
3	Consultorio Dental Brillante	44500
1	Clínica Dental Sonrisa Perfecta	30000