

START TRANSACTION;

```
CREATE DOMAIN tipo_membresia VARCHAR
    CHECK (VALUE IN ('Premium', 'Basico'));
CREATE DOMAIN tipo_genero VARCHAR
    CHECK (VALUE IN ('Masculino', 'Femenino'));
```

```
CREATE TABLE usuario(
    id_usuario SERIAL,
    correo_electronico VARCHAR NOT NULL UNIQUE,
    nombre VARCHAR NOT NULL,
    nacionalidad VARCHAR NOT NULL,
    domicilio VARCHAR NOT NULL ,
    genero tipo_genero NOT NULL ,
    tipo tipo_membresia NOT NULL ,
    CONSTRAINT pk_usuario PRIMARY KEY (id_usuario)
);
```

```
CREATE TABLE tarjeta(
    id_tarjeta SERIAL,
    num_tarjeta VARCHAR NOT NULL,
    id_usuario INT NOT NULL,
    CONSTRAINT fk_tarjeta_usuario FOREIGN KEY (id_usuario ) REFERENCES
public.usuario (id_usuario)
    ON UPDATE CASCADE ON DELETE NO ACTION,
    CONSTRAINT pk_tarjeta PRIMARY KEY (id_tarjeta)
);
```

```
CREATE TABLE medio_de_pago (
    id_mediodepago SERIAL,
    codigo_op INT NOT NULL UNIQUE,
    cbu VARCHAR NOT NULL ,
    num_tarjeta VARCHAR NOT NULL ,
    monto INT NOT NULL ,
    CONSTRAINT pk_mediodepago PRIMARY KEY (id_mediodepago)
);
```

```

CREATE TABLE interprete(
    id_interprete INT NOT NULL,
    cant_contenido INT NOT NULL ,
    cant_usuario INT NOT NULL ,
    cant_suscriptores INT,
    CONSTRAINT fk_intusuario FOREIGN KEY (id_interprete) REFERENCES
public.usuario (id_usuario)
    ON UPDATE CASCADE ON DELETE NO ACTION,
    CONSTRAINT pk_interprete PRIMARY KEY (id_interprete)
);

```

```

CREATE TABLE consumidor (
    id_consumidor INT NOT NULL,
    intereses VARCHAR NOT NULL ,
    CONSTRAINT fk_consusuario FOREIGN KEY (id_consumidor) REFERENCES
public.usuario (id_usuario)
    ON UPDATE CASCADE ON DELETE NO ACTION,
    CONSTRAINT pk_consumidor PRIMARY KEY (id_consumidor)
);

```

```

CREATE TABLE genero (
    id_genero SERIAL,
    nombre VARCHAR NOT NULL ,
    CONSTRAINT pk_genero PRIMARY KEY (id_genero),
    UNIQUE (nombre)
);

```

```

CREATE TABLE lista(
    id_lista SERIAL,
    url_lista VARCHAR NOT NULL UNIQUE,
    tipo_lista VARCHAR NOT NULL ,
    cant_canciones INT NOT NULL ,
    nombre VARCHAR NOT NULL UNIQUE,
    privacidad VARCHAR NOT NULL ,
    id_genero INT NOT NULL ,
    id_usuario INT NOT NULL ,
    CONSTRAINT fk_listausuario FOREIGN KEY (id_usuario) REFERENCES
public.usuario (id_usuario)
    ON UPDATE CASCADE ON DELETE NO ACTION,
    CONSTRAINT fk_listagenero FOREIGN KEY (id_genero) REFERENCES public.genero
);

```

```

(id_genero)
        ON UPDATE CASCADE ON DELETE NO ACTION,
    CONSTRAINT pk_lista PRIMARY KEY (id_lista)
);

```

```

CREATE TABLE album (
    id_album SERIAL,
    cant_pistas INT NOT NULL ,
    antiguedad INT NOT NULL ,
    fecha_lanzamiento DATE NOT NULL,
    nombre VARCHAR NOT NULL ,
    id_interprete INT NOT NULL,
    CONSTRAINT fk_albuminterprete FOREIGN KEY (id_interprete) REFERENCES
public.interprete (id_interprete)
        ON UPDATE CASCADE ON DELETE CASCADE,
    CONSTRAINT pk_album PRIMARY KEY (id_album),
    UNIQUE (nombre, id_interprete)
);

```

```

CREATE DOMAIN tipo_contenido varchar
    CHECK (VALUE IN ('Musica', 'Podcast'));

```

```

CREATE TABLE contenido(
    id_contenido SERIAL,
    url VARCHAR NOT NULL UNIQUE,
    tipo tipo_contenido NOT NULL ,
    nombre VARCHAR NOT NULL ,
    duracion INT NOT NULL ,
    comentarios VARCHAR ,
    cant_likes INT ,
    reproduc_totales INT NOT NULL ,
    id_genero INT NOT NULL,
    id_album INT,
    id_interprete INT NOT NULL,
    CONSTRAINT fk_continterprete FOREIGN KEY (id_interprete) REFERENCES
public.interprete (id_interprete)
        ON UPDATE CASCADE ON DELETE NO ACTION,
    CONSTRAINT fk_contalbum FOREIGN KEY (id_album) REFERENCES public.album
(id_album)
        ON UPDATE CASCADE ON DELETE NO ACTION,
    CONSTRAINT fk_contgenero FOREIGN KEY (id_genero) REFERENCES public.genero

```

```

(id_genero)
    ON UPDATE CASCADE ON DELETE NO ACTION,
    CONSTRAINT pk_idcontenido PRIMARY KEY (id_contenido)
);

```

```

CREATE TABLE es_suscriptor (
    es_seguido INT,
    sigue_a INT,
    id_usuario INT NOT NULL,
    CONSTRAINT fk_es_seguido FOREIGN KEY (id_usuario) REFERENCES
public.usuario (id_usuario)
    ON UPDATE CASCADE ON DELETE NO ACTION,
    CONSTRAINT fk_sigue_a FOREIGN KEY (id_usuario) REFERENCES public.usuario
(id_usuario)
    ON UPDATE CASCADE ON DELETE NO ACTION,
    CONSTRAINT pk_es_sigue PRIMARY KEY (es_seguido,sigue_a)
);

```

```

CREATE TABLE asociados (
    esta_asociado INT,
    es_asociado INT,
    id_usuario INT NOT NULL,
    CONSTRAINT fk_esta_asociado FOREIGN KEY (id_usuario) REFERENCES
public.usuario (id_usuario)
    ON UPDATE CASCADE ON DELETE NO ACTION,
    CONSTRAINT fk_es_asociado FOREIGN KEY (id_usuario) REFERENCES
public.usuario (id_usuario)
    ON UPDATE CASCADE ON DELETE NO ACTION,
    CONSTRAINT pk_asociado PRIMARY KEY (esta_asociado, es_asociado)
);

```

```

CREATE TABLE es_subgenero (
    id_genero INT NOT NULL,
    id_subgenero INT NOT NULL,
    CONSTRAINT fk_genero FOREIGN KEY (id_genero) REFERENCES public.genero
(id_genero)
    ON UPDATE CASCADE ON DELETE NO ACTION,
    CONSTRAINT fk_subgenero FOREIGN KEY (id_subgenero) REFERENCES
public.genero (id_genero)
    ON UPDATE CASCADE ON DELETE NO ACTION,

```

```
        CONSTRAINT pk_essubgenero PRIMARY KEY (id_genero)
    );
```

```
CREATE TABLE posee (
    id_posee SERIAL,
    id_usuario INT NOT NULL,
    CONSTRAINT fk_poseeusuario FOREIGN KEY (id_usuario) REFERENCES
public.usuario (id_usuario)
        ON UPDATE CASCADE ON DELETE NO ACTION,
    CONSTRAINT pk_posee PRIMARY KEY (id_posee)
);
```

```
CREATE TABLE es_agregado (
    id_agregado SERIAL,
    id_lista INT NOT NULL,
    id_contenido INT NOT NULL,
    CONSTRAINT fk_agregacontenido FOREIGN KEY (id_contenido) REFERENCES
public.contenido (id_contenido)
        ON UPDATE CASCADE ON DELETE NO ACTION,
    CONSTRAINT fk_agregalista FOREIGN KEY (id_lista) REFERENCES public.lista
(id_lista)
        ON UPDATE CASCADE ON DELETE NO ACTION,
    CONSTRAINT pk_agregado PRIMARY KEY (id_agregado),
    UNIQUE (id_lista, id_contenido)
);
```

```
CREATE TABLE abona (
    id_abona SERIAL,
    id_usuario INT NOT NULL,
    id_mediodepago INT NOT NULL,
    CONSTRAINT fk_abonausuario FOREIGN KEY (id_usuario) REFERENCES
public.usuario (id_usuario)
        ON UPDATE CASCADE ON DELETE NO ACTION,
    CONSTRAINT fk_abonamediodepago FOREIGN KEY (id_mediodepago)
REFERENCES public.medio_de_pago (id_mediodepago)
        ON UPDATE CASCADE ON DELETE NO ACTION,
    CONSTRAINT pk_abona PRIMARY KEY (id_abona),
    UNIQUE (id_usuario, id_mediodepago)
);
```

```
CREATE TABLE califica(  
    id_califica SERIAL,  
    id_usuario INT NOT NULL,  
    id_contenido INT NOT NULL,  
    CONSTRAINT fk_califusuario FOREIGN KEY (id_usuario) REFERENCES  
public.usuario (id_usuario)  
        ON UPDATE CASCADE ON DELETE NO ACTION,  
    CONSTRAINT fk_califcontenido FOREIGN KEY (id_contenido) REFERENCES  
public.contenido (id_contenido)  
        ON UPDATE CASCADE ON DELETE NO ACTION,  
    CONSTRAINT pk_califica PRIMARY KEY (id_califica),  
    UNIQUE (id_usuario, id_contenido)  
);  
  
COMMIT;
```