

Tarea 1.2: Diseño de las tablas en PostgreSQL:

- Crear las tablas en PostgreSQL utilizando el lenguaje SQL, definiendo los tipos de datos adecuados para cada columna (enteros, texto, fechas, etc.).
- Establecer las claves primarias y foráneas para garantizar la integridad de los datos.
 - Primero creamos la base de datos:

The screenshot shows the PostgreSQL query editor interface. The query editor has a toolbar with icons for saving, generating queries, formatting SQL, and viewing code snippets. Below the toolbar, the query editor shows the following SQL code:

```
1 CREATE DATABASE "creativaonline"
2 WITH
3 ENCODING = "UTF-8"
4 ;
```

Below the query editor, the 'Mensaje' (Message) tab is selected, showing the execution results:

Consulta	Mensaje	Consulta de h...	Tiempo de rec...
CREATE DATABASE "creativaonline" WITH ENCODING = "...	OK	0,462s	0,000s

- Ahora creamos cada una de las tablas de la base de datos:

The screenshot shows the PostgreSQL query editor interface. The query editor has a toolbar with icons for saving, generating queries, formatting SQL, and viewing code snippets. Below the toolbar, the query editor shows the following SQL code:

```
1 -- Tabla de Clientes
2 CREATE TABLE clientes (
3   id_cliente SERIAL PRIMARY KEY,
4   nombre VARCHAR(255) NOT NULL,
5   razon_social VARCHAR(255),
6   contacto_principal VARCHAR(255) NOT NULL,
7   telefono VARCHAR(20),
8   correo VARCHAR(255) UNIQUE NOT NULL
9 );
```

Below the query editor, the 'Mensaje' (Message) tab is selected, showing the execution results:

Consulta	Mensaje	Consulta de h...	Tiempo de rec...
-- Tabla de Clientes CREATE TABLE clientes (id_cliente SER...	OK	0,024s	0,000s

Objetos * Sin título - Consulta

Guardar Generador de Consultas Embellecer SQL Fragmento de código

PostgreSQL creativaonline public Ejecutar Detener Explicar

```

1  -- Tabla de Proyectos
2  CREATE TABLE proyectos (
3      id_proyecto SERIAL PRIMARY KEY,
4      nombre VARCHAR(255) NOT NULL,
5      descripcion TEXT,
6      id_cliente INT NOT NULL,
7      fecha_inicio DATE NOT NULL,
8      fecha_fin DATE,
9      presupuesto DECIMAL(15,2),
10     estado VARCHAR(50) CHECK (estado IN ('Planificado', 'En Curso', 'Completado', 'Cancelado')),
11     FOREIGN KEY (id_cliente) REFERENCES clientes(id_cliente) ON DELETE CASCADE
12 );

```

Mensaje Resumen

Consulta procesada: 1 Hora de inicio: 2025-02-16 16:18:44
Satisfactorio: 1 Hora de finalización: 2025-02-16 16:18:44
Error: 0 Tiempo transcurrido: 0.038s ☐ Mostrar sólo los errores

Consulta	Mensaje	Consulta de h...	Tiempo de rec...
-- Tabla de Proyectos CREATE TABLE proyectos (id_proye...	OK	0,008s	0,000s

Objetos * Sin título - Consulta

Guardar Generador de Consultas Embellecer SQL Fragmento de código

PostgreSQL creativaonline public Ejecutar Detener Explicar

```


1  -- Tabla de Recursos (Personal)
2  CREATE TABLE recursos (
3      id_recurso SERIAL PRIMARY KEY,
4      nombre VARCHAR(255) NOT NULL,
5      apellidos VARCHAR(255) NOT NULL,
6      rol VARCHAR(100) NOT NULL,
7      disponibilidad BOOLEAN DEFAULT TRUE,
8      habilidades TEXT
9  );




```

Mensaje Resumen

Consulta procesada: 1 Hora de inicio: 2025-02-16 16:19:34
Satisfactorio: 1 Hora de finalización: 2025-02-16 16:19:34
Error: 0 Tiempo transcurrido: 0.052s ☐ Mostrar sólo los errores

Consulta	Mensaje	Consulta de h...	Tiempo de rec...
-- Tabla de Recursos (Personal) CREATE TABLE recursos (i...	OK	0,024s	0,000s

Objetos  Sin título - Consulta

Guardar  Generador de Consultas  Embellecer SQL  Fragmento de código

PostgreSQL creativaonline public Ejecutar Detener Explicar

```


1  -- Tabla de Tareas
2  CREATE TABLE tareas (
3      id_tarea SERIAL PRIMARY KEY,
4      nombre VARCHAR(255) NOT NULL,
5      descripcion TEXT,
6      id_proyecto INT NOT NULL,
7      id_responsable INT,
8      fecha_inicio DATE NOT NULL,
9      fecha_fin DATE,
10     estado VARCHAR(50) CHECK (estado IN ('Pendiente', 'En Progreso', 'Completada', 'Bloqueada')),
11     prioridad VARCHAR(20) CHECK (prioridad IN ('Baja', 'Media', 'Alta')),
12     FOREIGN KEY (id_proyecto) REFERENCES proyectos(id_proyecto) ON DELETE CASCADE,
13     FOREIGN KEY (id_responsable) REFERENCES recursos(id_recurso) ON DELETE SET NULL
14 );
15 -- Tabla de Tareas
16 CREATE TABLE tareas (
17     id_tarea SERIAL PRIMARY KEY,
18     nombre VARCHAR(255) NOT NULL,
19     descripcion TEXT,
20     id_proyecto INT NOT NULL,
21     id_responsable INT,
22     fecha_inicio DATE NOT NULL,
23     fecha_fin DATE,
24     estado VARCHAR(50) CHECK (estado IN ('Pendiente', 'En Progreso', 'Completada', 'Bloqueada')),
25     prioridad VARCHAR(20) CHECK (prioridad IN ('Baja', 'Media', 'Alta')),
26     FOREIGN KEY (id_proyecto) REFERENCES proyectos(id_proyecto) ON DELETE CASCADE,
27     FOREIGN KEY (id_responsable) REFERENCES recursos(id_recurso) ON DELETE SET NULL
28 );




```

Mensaje Resumen

Consulta procesada: 1 Hora de inicio: 2025-02-16 16:20:10
Satisfactorio: 1 Hora de finalización: 2025-02-16 16:20:10
Error: 0 Tiempo transcurrido: 0.045s ☐ Mostrar sólo los errores

Consulta	Mensaje	Consulta de h...	Tiempo de rec...
-- Tabla de Tareas CREATE TABLE tareas (id_tarea SERIAL P...	OK	0,016s	0,000s

Objetos  Sin título - Consulta

Guardar  Generador de Consultas  Embellecer SQL  Fragmento de código

PostgreSQL creativaonline public Ejecutar Detener Explicar

```

1  -- Tabla de Documentos
2  CREATE TABLE documentos (
3      id_documento SERIAL PRIMARY KEY,
4      nombre VARCHAR(255) NOT NULL,
5      id_proyecto INT NOT NULL,
6      version INT DEFAULT 1,
7      url TEXT NOT NULL,
8      FOREIGN KEY (id_proyecto) REFERENCES proyectos(id_proyecto) ON DELETE CASCADE
9  );
10

```

Mensaje Resumen

Consulta procesada: 1 Hora de inicio: 2025-02-16 16:21:39
Satisfactorio: 1 Hora de finalización: 2025-02-16 16:21:39
Error: 0 Tiempo transcurrido: 0.055s ☐ Mostrar sólo los errores

Consulta	Mensaje	Consulta de h...	Tiempo de rec...
-- Tabla de Documentos CREATE TABLE documentos (id_...	OK	0,027s	0,000s

Objetos * Sin título - Consulta

Guardar Generador de Consultas Embellecer SQL Fragmento de código

PostgreSQL creativaonline public Ejecutar Detener Explicar

```

1 -- Tabla de Seguimiento de Horas
2 CREATE TABLE seguimiento_horas (
3     id_registro SERIAL PRIMARY KEY,
4     id_recurso INT NOT NULL,
5     id_tarea INT NOT NULL,
6     horas_trabajadas DECIMAL(5,2) NOT NULL,
7     fecha_registro DATE NOT NULL DEFAULT CURRENT_DATE,
8     FOREIGN KEY (id_recurso) REFERENCES recursos(id_recurso) ON DELETE CASCADE,
9     FOREIGN KEY (id_tarea) REFERENCES tareas(id_tarea) ON DELETE CASCADE
10 );

```

Mensaje Resumen

Consulta procesada: 1 Hora de inicio: 2025-02-16 16:22:11
 Satisfactorio: 1 Hora de finalización: 2025-02-16 16:22:11
 Error: 0 Tiempo transcurrido: 0.062s ☐ Mostrar sólo los errores

Consulta	Mensaje	Consulta de h...	Tiempo de rec...
-- Tabla de Seguimiento de Horas CREATE TABLE segui...	OK	0,033s	0,000s

Objetos * Sin título - Consulta

Guardar Generador de Consultas Embellecer SQL Fragmento de código

PostgreSQL creativaonline public Ejecutar Detener Explicar

```

1 -- Tabla de Comunicaciones (Chat interno)
2 CREATE TABLE comunicaciones (
3     id_mensaje SERIAL PRIMARY KEY,
4     id_remitente INT NOT NULL,
5     id_destinatario INT NOT NULL,
6     mensaje TEXT NOT NULL,
7     fecha_envio TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
8     FOREIGN KEY (id_remitente) REFERENCES recursos(id_recurso) ON DELETE CASCADE,
9     FOREIGN KEY (id_destinatario) REFERENCES recursos(id_recurso) ON DELETE CASCADE
10 );

```

Mensaje Resumen

Consulta procesada: 1 Hora de inicio: 2025-02-16 16:23:27
 Satisfactorio: 1 Hora de finalización: 2025-02-16 16:23:27
 Error: 0 Tiempo transcurrido: 0.041s ☐ Mostrar sólo los errores

Consulta	Mensaje	Consulta de h...	Tiempo de rec...
-- Tabla de Comunicaciones (Chat interno) CREATE TABLE...	OK	0,012s	0,000s

c) Crear índices para mejorar el rendimiento de las consultas.

Objetos * Sin título - Consulta

Guardar Generador de Consultas Embellecer SQL Fragmento de código

PostgreSQL creativaonline public Ejecutar Detener Explicar

```

1 -- Índices para mejorar el rendimiento
2 CREATE INDEX idx_proyectos_cliente ON proyectos(id_cliente);
3 CREATE INDEX idx_tareas_proyecto ON tareas(id_proyecto);
4 CREATE INDEX idx_tareas_responsable ON tareas(id_responsable);
5 CREATE INDEX idx_documentos_proyecto ON documentos(id_proyecto);
6 CREATE INDEX idx_horas_tarea ON seguimiento_horas(id_tarea);

```

Mensaje Resumen

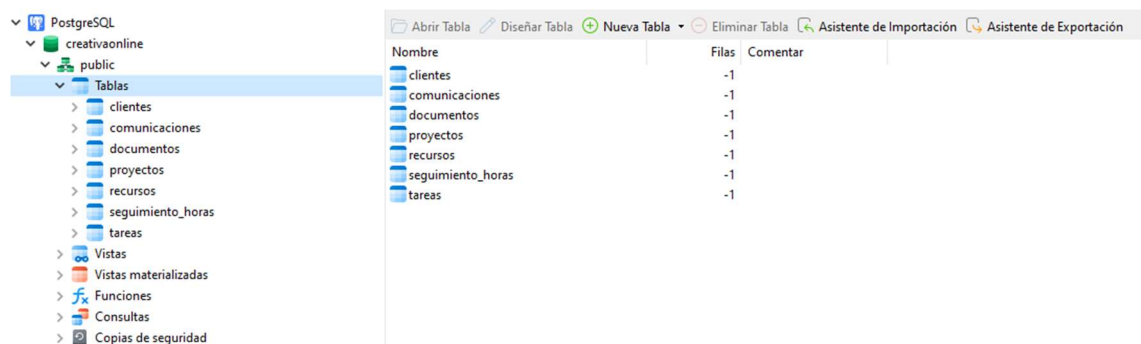
Consulta procesada: 5 Hora de inicio: 2025-02-16 16:24:32
 Satisfactorio: 5 Hora de finalización: 2025-02-16 16:24:33
 Error: 0 Tiempo transcurrido: 0.042s ☐ Mostrar sólo los errores

Consulta	Mensaje	Consulta de h...	Tiempo de rec...
-- Índices para mejorar el rendimiento CREATE INDEX idx_...	OK	0,002s	0,000s
CREATE INDEX idx_tareas_proyecto ON tareas(id_proyecto)	OK	0,001s	0,000s
CREATE INDEX idx_tareas_responsable ON tareas(id_respo...	OK	0,001s	0,000s
CREATE INDEX idx_documentos_proyecto ON documento...	OK	0,001s	0,000s
CREATE INDEX idx_horas_tarea ON seguimiento_horas(id_...	OK	0,001s	0,000s

Observaciones:

- a) Para el diseño utilice tipo SERIAL para las claves primarias, obteniendo un id auto incrementable.
- b) Se utilizó ON DELETE CASCADE en las claves foráneas donde la eliminación de un elemento padre deba reflejarse en sus hijos.
- c) Se utilizó ON DELETE SET NULL en las tareas cuando se necesario resguardar un historial si un elemento es borrado.
- d) Se crearon índices en las claves foráneas para mejorar la eficiencia en las búsquedas y filtrados.

Base de datos con sus tablas finalmente creada:



Tarea 1.3: Normalización de las tablas:

- a) Aplicar las reglas de normalización (primera, segunda y tercera forma normal) para asegurar la consistencia y reducir la redundancia de los datos.
- b) Tarea 1.4: Creación de vistas:
- c) Crear vistas para simplificar consultas complejas y proporcionar acceso restringido a ciertos datos.

FASE 2 - Inserción, Consulta y Eliminación de Datos

Tarea 2.1: Inserción de datos:

- a) Crear scripts SQL para insertar grandes volúmenes de datos en las tablas.
- b) Utilizar comandos de inserción múltiple para mejorar la eficiencia.
- c) Validar los datos antes de insertarlos para garantizar la integridad.