

1.

Las entidades principales involucradas en el ejercicio son:

1. Oferta de Empleo: Representa las oportunidades laborales publicadas por las empresas. Contiene información como el título de la oferta, la descripción del puesto, la empresa que la publica y la fecha de publicación.
2. Aspirante: Representa a las personas que buscan empleo y se postulan para las ofertas disponibles. Contiene información como el nombre, apellido, correo electrónico, teléfono y dirección del aspirante.
3. Candidatura: Representa la relación entre una oferta de empleo y un aspirante. Contiene información sobre el estado de la candidatura, que puede ser pendiente, en proceso, rechazada o contratada. Esta entidad permite asociar un aspirante a una oferta de empleo específica y realizar un seguimiento del proceso de selección.
4. Entrevista: Representa las entrevistas programadas para las candidaturas seleccionadas. Contiene información sobre la fecha, lugar y observaciones relacionadas con la entrevista realizada.

Estas entidades principales forman la base para la gestión de la información relacionada con la búsqueda de empleo de los aspirantes.

2. Atributos de las entidades (1 punto): Para cada entidad, enumera los atributos relevantes. Por ejemplo: en "Usuario" podríamos tener "ID", "Nombre", "Apellido", "Email", "Contraseña", etc. En "Oferta de Empleo" podríamos tener "ID", "Título", "Descripción", "Ubicación", etc.

1. OfertasDeEmpleo:

- IDOferta
- Titulo
- Descripcion
- Empresa
- FechaPublicacion
- Ciudad

2. Aspirantes:

- IDAspirante
- Nombre
- Apellido
- Edad
- Email

- Telefono
- Dirección

3. Candidaturas:

- IDCandidatura
- IDOferta
- IDAspirante
- Estado

3. Relaciones entre las entidades (2 puntos): Determina las relaciones entre las entidades identificadas. ¿Cómo se relacionan las entidades "Usuario", "Oferta de Empleo" e "Inscripción"? Indica las claves primarias y las claves foráneas necesarias en cada tabla

Basado en la descripción del ejercicio, las entidades identificadas son "OfertasDeEmpleo", "Aspirantes" y "Candidaturas". No se mencionan las entidades "Usuario" ni "Inscripción" en el contexto proporcionado.

A continuación, se describen las relaciones entre las entidades "OfertasDeEmpleo", "Aspirantes" y "Candidaturas", y se indican las claves primarias y las claves foráneas necesarias en cada tabla:

1. OfertasDeEmpleo:

- Clave primaria: IDOferta (PK)

2. Aspirantes:

- Clave primaria: IDAspirante (PK)

3. Candidaturas:

- Clave primaria: IDCandidatura (PK)
- Clave foránea: IDOferta (FK referenciando a la tabla "OfertasDeEmpleo")
- Clave foránea: IDAspirante (FK referenciando a la tabla "Aspirantes")

La entidad "Candidaturas" se utiliza para establecer la relación entre las entidades "OfertasDeEmpleo" y "Aspirantes". La clave foránea "IDOferta" en la tabla "Candidaturas" referencia la clave primaria "IDOferta" en la tabla "OfertasDeEmpleo", y la clave foránea "IDAspirante" en la tabla "Candidaturas" referencia la clave primaria "IDAspirante" en la tabla "Aspirantes".

Estas relaciones permiten asociar una oferta de empleo específica con uno o varios aspirantes a través de la tabla "Candidaturas", y realizar un seguimiento del estado de cada candidatura en el proceso de búsqueda de empleo.

4. Formas normales de Codd (1 punto): Explica qué formas normales de Codd aplicarías para garantizar la integridad de los datos y evitar redundancias y anomalías en la base de datos.

Las tablas propuestas cumplen con las siguientes formas normales de Codd:

1. Primera Forma Normal (1FN): Todas las tablas tienen campos atómicos, es decir, no contienen valores multivaluados o repetidos. Cada columna de una tabla contiene un solo valor.
2. Segunda Forma Normal (2FN): En las tablas proporcionadas, no hay dependencias parciales, lo que significa que cada columna depende completamente de la clave primaria. No hay campos que dependan solo de parte de la clave primaria.
3. Tercera Forma Normal (3FN): Las tablas están en tercera forma normal ya que no existen dependencias transitivas. Los campos no clave dependen únicamente de la clave primaria y no hay dependencias indirectas entre los campos no clave.

En resumen, las tablas propuestas cumplen con las formas normales de Codd hasta la Tercera Forma Normal (1FN, 2FN y 3FN). Esto ayuda a garantizar la integridad de los datos y minimizar la redundancia y las anomalías en la base de datos.

5. Diseño de las tablas (1 punto): Diseña las tablas correspondientes a cada entidad, asegurándote de incluir los atributos, las claves primarias y las claves foráneas necesarias. Indica también los tipos de datos adecuados para cada atributo.

Aquí tienes la última propuesta de diseño, incluyendo los tipos de datos adecuados para cada atributo:

1. Tabla "OfertasDeEmpleo":

- IDOferta (clave primaria) [Tipo de datos: INTEGER]
- Titulo [Tipo de datos: VARCHAR]
- Descripcion [Tipo de datos: VARCHAR]
- Empresa [Tipo de datos: VARCHAR]
- FechaPublicacion [Tipo de datos: DATE]
- Ciudad [Tipo de datos: VARCHAR]

2. Tabla "Aspirantes":

- IDAspirante (clave primaria) [Tipo de datos: INTEGER]
- Nombre [Tipo de datos: VARCHAR]
- Apellido [Tipo de datos: VARCHAR]
- Edad [Tipo de datos: INTEGER]
- Email [Tipo de datos: VARCHAR]
- Telefono [Tipo de datos: VARCHAR]
- Dirección [Tipo de datos: VARCHAR]

3. Tabla "Candidaturas":

- IDCandidatura (clave primaria) [Tipo de datos: INTEGER]
- IDOferta (clave foránea referenciando a la tabla "OfertasDeEmpleo") [Tipo de datos: INTEGER]
- IDAspirante (clave foránea referenciando a la tabla "Aspirantes") [Tipo de datos: INTEGER]
- Estado [Tipo de datos: VARCHAR]

En esta propuesta, se han especificado los tipos de datos adecuados para cada atributo en función de la naturaleza de los datos que se almacenan. Ten en cuenta que los tipos de datos pueden variar según la implementación específica de la base de datos Oracle. Asegúrate de adaptar los tipos de datos según tus necesidades y las características de tu sistema.

6. Restricciones y claves primarias (1 punto): Define las restricciones y las claves primarias en las tablas para garantizar la integridad de los datos.

1.

Tabla "OfertasDeEmpleo":

- Clave primaria: IDOferta (PK) - Es recomendable definir esta columna como clave primaria para asegurar que cada oferta de empleo tenga un identificador único.

2. Tabla "Aspirantes":

- Clave primaria: IDAspirante (PK) - Esta columna se define como clave primaria para garantizar que cada aspirante tenga un identificador único.

3. Tabla "Candidaturas":

- Clave primaria: IDCandidatura (PK) - Se establece como clave primaria para asegurar que cada candidatura tenga un identificador único.

Además de las claves primarias, puedes establecer otras restricciones para garantizar la integridad de los datos. Algunas restricciones recomendadas pueden incluir:

- Restricciones de integridad referencial: Puedes establecer restricciones de integridad referencial para las claves foráneas "IDOferta" y "IDAspirante" en la tabla "Candidaturas" para asegurarte de que siempre haga referencia a registros existentes en las tablas "OfertasDeEmpleo" y "Aspirantes", respectivamente.
- Restricciones de dominio: Puedes establecer restricciones de dominio para los atributos, como por ejemplo, asegurarte de que la edad sea un número positivo, que los correos electrónicos tengan un formato válido, etc.

Estas son solo algunas recomendaciones generales, pero las restricciones y claves primarias pueden variar según las necesidades específicas y las características del sistema.

7. Inserción de datos de ejemplo (1 punto): Inserta algunos datos de ejemplo en las tablas para verificar el diseño y la funcionalidad. Utiliza sentencias SQL INSERT para este propósito.

-- Inserción de datos en la tabla "OfertasDeEmpleo"

```
INSERT INTO OfertasDeEmpleo (Titulo, Descripcion, Empresa, FechaPublicacion, Ciudad)
```

```
VALUES ('Desarrollador Full Stack', 'Empresa XYZ busca un desarrollador Full Stack con experiencia en tecnologías web.', 'Empresa XYZ', TO_DATE('2023-06-01', 'YYYY-MM-DD'), 'Ciudad A');
```

```
INSERT INTO OfertasDeEmpleo (Titulo, Descripcion, Empresa, FechaPublicacion, Ciudad)
```

```
VALUES ('Analista de Datos', 'Empresa ABC busca un analista de datos con conocimientos en minería de datos y estadística.', 'Empresa ABC', TO_DATE('2023-06-05', 'YYYY-MM-DD'), 'Ciudad B');
```

```
INSERT INTO OfertasDeEmpleo (Titulo, Descripcion, Empresa, FechaPublicacion, Ciudad)
```

```
VALUES ('Diseñador Gráfico', 'Empresa DEF busca un diseñador gráfico con experiencia en Photoshop e Illustrator.', 'Empresa DEF', TO_DATE('2023-06-10', 'YYYY-MM-DD'), 'Ciudad C');
```

```
INSERT INTO OfertasDeEmpleo (Titulo, Descripcion, Empresa, FechaPublicacion, Ciudad)
```

```
VALUES ('Asistente Administrativo', 'Empresa GHI busca un asistente administrativo con habilidades organizativas.', 'Empresa GHI', TO_DATE('2023-06-15', 'YYYY-MM-DD'), 'Ciudad D');
```

```
INSERT INTO OfertasDeEmpleo (Titulo, Descripcion, Empresa, FechaPublicacion, Ciudad)
VALUES ('Ingeniero de Software', 'Empresa JKL busca un ingeniero de software con experiencia
en desarrollo de aplicaciones móviles.', 'Empresa JKL', TO_DATE('2023-06-20', 'YYYY-MM-DD'),
'Ciudad E');
```

-- Inserción de datos en la tabla "Aspirantes"

```
INSERT INTO Aspirantes (Nombre, Apellido, Edad, Email, Telefono, Direccion)
VALUES ('Juan', 'Pérez', 30, 'juan@example.com', '1234567890', 'Calle Principal 123');
```

```
INSERT INTO Aspirantes (Nombre, Apellido, Edad, Email, Telefono, Direccion)
VALUES ('María', 'Gómez', 25, 'maria@example.com', '9876543210', 'Avenida Central 456');
```

```
INSERT INTO Aspirantes (Nombre, Apellido, Edad, Email, Telefono, Direccion)
VALUES ('Carlos', 'López', 28, 'carlos@example.com', '5551234567', 'Plaza Mayor 789');
```

```
INSERT INTO Aspirantes (Nombre, Apellido, Edad, Email, Telefono, Direccion)
VALUES ('Laura', 'Fernández', 35, 'laura@example.com', '7779876543', 'Calle Secundaria 321');
```

```
INSERT INTO Aspirantes (Nombre, Apellido, Edad, Email, Telefono, Direccion)
VALUES ('Pedro', 'Rodríguez', 27, 'pedro@example.com', '1112223334', 'Avenida Principal
567');
```

```
INSERT INTO Aspirantes (Nombre, Apellido, Edad, Email, Telefono, Direccion)
VALUES ('Ana', 'Martínez', 29, 'ana@example.com', '8887776665', 'Calle Central 890');
```

```
INSERT INTO Aspirantes (Nombre, Apellido, Edad, Email, Telefono, Direccion)
VALUES ('Luis', 'García', 32, 'luis@example.com', '3334445556', 'Plaza Secundaria 123');
```

```
INSERT INTO Aspirantes (Nombre, Apellido, Edad, Email, Telefono, Direccion)
VALUES ('Sofía', 'Vargas', 26, 'sofia@example.com', '9998887778', 'Avenida Secundaria 456');
```

```
INSERT INTO Aspirantes (Nombre, Apellido, Edad, Email, Telefono, Direccion)
```

```
VALUES ('Jorge', 'Hernández', 31, 'jorge@example.com', '2223334447', 'Calle Mayor 789');
```

```
INSERT INTO Aspirantes (Nombre, Apellido, Edad, Email, Telefono, Direccion)
```

```
VALUES ('Marta', 'Luna', 24, 'marta@example.com', '6667778889', 'Plaza Central 012');
```

```
-- Inserción de datos en la tabla "Candidaturas"
```

```
INSERT INTO Candidaturas (IDOferta, IDAspirante, Estado)
```

```
VALUES (1, 1, 'Pendiente');
```

```
INSERT INTO Candidaturas (IDOferta, IDAspirante, Estado)
```

```
VALUES (1, 2, 'Preseleccionado');
```

```
INSERT INTO Candidaturas (IDOferta, IDAspirante, Estado)
```

```
VALUES (2, 3, 'Finalista');
```

```
INSERT INTO Candidaturas (IDOferta, IDAspirante, Estado)
```

```
VALUES (2, 4, 'Seleccionado');
```

```
INSERT INTO Candidaturas (IDOferta, IDAspirante, Estado)
```

```
VALUES (3, 5, 'Pendiente');
```

```
INSERT INTO Candidaturas (IDOferta, IDAspirante, Estado)
```

```
VALUES (3, 6, 'Preseleccionado');
```

```
INSERT INTO Candidaturas (IDOferta, IDAspirante, Estado)
```

```
VALUES (4, 7, 'Finalista');
```

```
INSERT INTO Candidaturas (IDOferta, IDAspirante, Estado)
```

```
VALUES (4, 8, 'Seleccionado');
```

```
INSERT INTO Candidaturas (IDOferta, IDAspirante, Estado)
```

```
VALUES (5, 9, 'Pendiente');
```

```
INSERT INTO Candidaturas (IDOferta, IDAspirante, Estado)
```

```
VALUES (5, 10, 'Preseleccionado');
```

```
INSERT INTO Candidaturas (IDOferta, IDAspirante, Estado)
```

```
VALUES (1, 3, 'Finalista');
```

```
INSERT INTO Candidaturas (IDOferta, IDAspirante, Estado)
```

```
VALUES (2, 5, 'Seleccionado');
```

```
INSERT INTO Candidaturas (IDOferta, IDAspirante, Estado)
```

```
VALUES (3, 7, 'Pendiente');
```

```
INSERT INTO Candidaturas (IDOferta, IDAspirante, Estado)
```

```
VALUES (4, 9, 'Preseleccionado');
```

```
INSERT INTO Candidaturas (IDOferta, IDAspirante, Estado)
```

```
VALUES (5, 1, 'Finalista');
```

```
INSERT INTO Candidaturas (IDOferta, IDAspirante, Estado)
```

```
VALUES (1, 4, 'Seleccionado');
```

```
INSERT INTO Candidaturas (IDOferta, IDAspirante, Estado)
```

```
VALUES (2, 6, 'Pendiente');
```

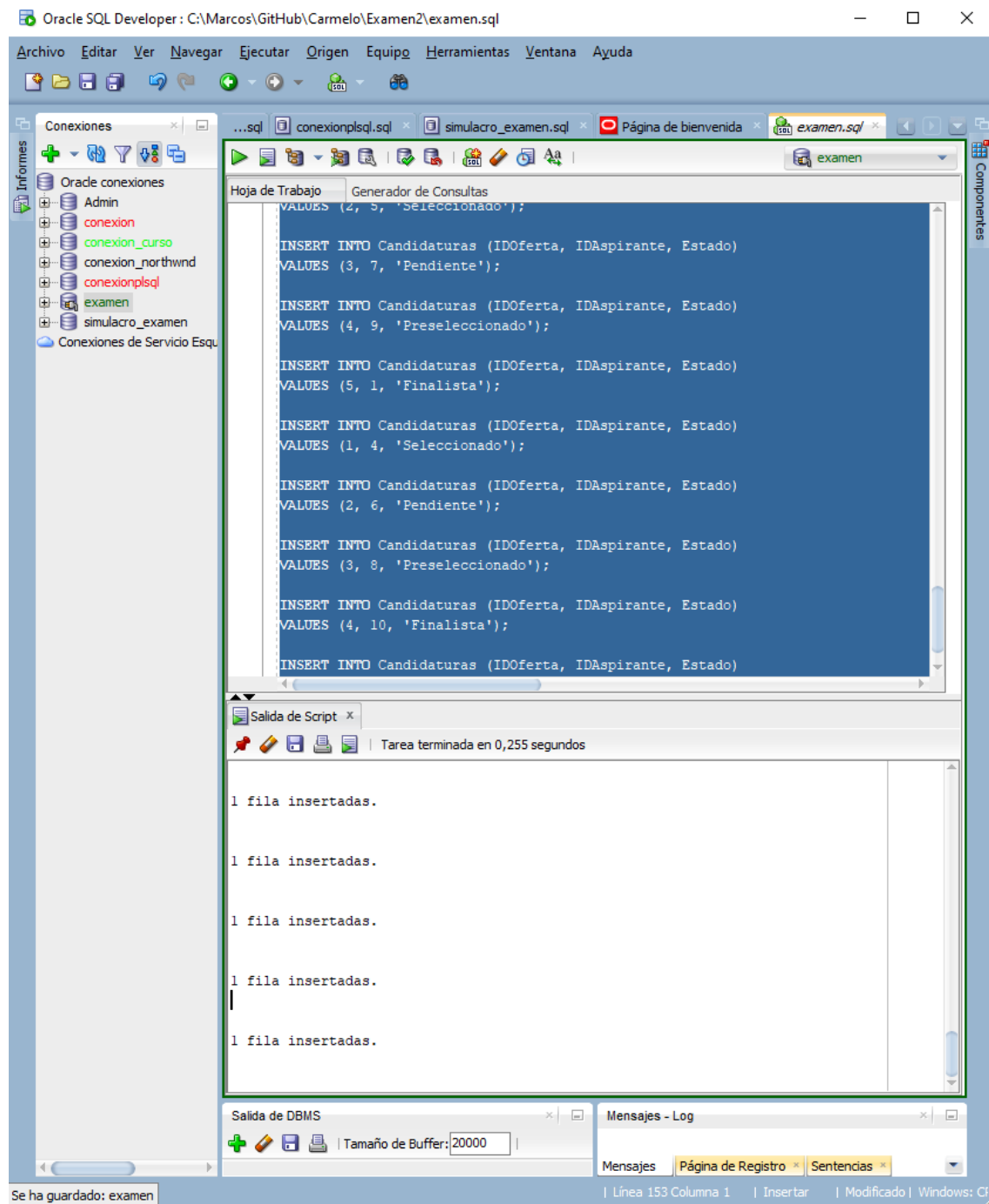
```
INSERT INTO Candidaturas (IDOferta, IDAspirante, Estado)
```

```
VALUES (3, 8, 'Preseleccionado');
```

```
INSERT INTO Candidaturas (IDOferta, IDAspirante, Estado)
```

```
VALUES (4, 10, 'Finalista');
```

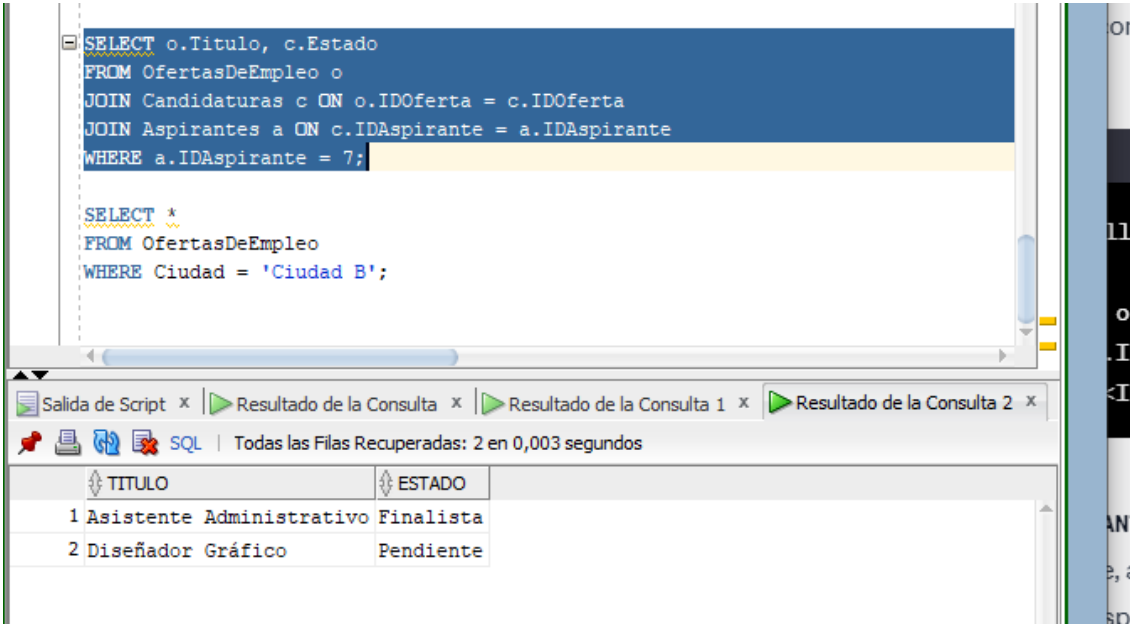

INSERT INTO Candidaturas (IDOferta, IDAspirante, Estado)
VALUES (5, 2, 'Seleccionado');



8. Consultas (1 punto): Proporciona una consulta SQL que permita obtener información relevante de la base de datos. Por ejemplo, podrías consultar las inscripciones de un usuario en particular o las ofertas de empleo disponibles en una ubicación específica.

```
SELECT o.Titulo, c.Estado
FROM OfertasDeEmpleo o
JOIN Candidaturas c ON o.IDOferta = c.IDOferta
JOIN Aspirantes a ON c.IDAspirante = a.IDAspirante
WHERE a.IDAspirante = 7;
```

```
SELECT *
FROM OfertasDeEmpleo
WHERE Ciudad = 'Ciudad B';
```



The screenshot shows a SQL query editor with two queries. The first query is highlighted in blue and the second in yellow. Below the queries, there is a table with two columns: TITULO and ESTADO. The table contains two rows of data.

```
SELECT o.Titulo, c.Estado
FROM OfertasDeEmpleo o
JOIN Candidaturas c ON o.IDOferta = c.IDOferta
JOIN Aspirantes a ON c.IDAspirante = a.IDAspirante
WHERE a.IDAspirante = 7;

SELECT *
FROM OfertasDeEmpleo
WHERE Ciudad = 'Ciudad B';
```

	TITULO	ESTADO
1	Asistente Administrativo	Finalista
2	Diseñador Gráfico	Pendiente

9. Verificación y ajuste del diseño (1 punto): Verifica que el diseño de la base de datos sea correcto y realiza ajustes si es necesario para mejorar la eficiencia y la funcionalidad.

```
SELECT o.Titulo, c.Estado
FROM OfertasDeEmpleo o
JOIN Candidaturas c ON o.IDOferta = c.IDOferta
JOIN Aspirantes a ON c.IDAspirante = a.IDAspirante
WHERE a.IDAspirante = 7;
```

Esta consulta devolverá el título de las ofertas de empleo y el estado de las candidaturas del aspirante especificado.

```
SELECT *
FROM OfertasDeEmpleo
WHERE Ciudad = 'Ciudad B';
```

Esta consulta devolverá todas las columnas de la tabla "OfertasDeEmpleo" para las ofertas de empleo disponibles en la ciudad especificada.

The screenshot shows a SQL query editor with two queries. The first query is highlighted in blue and the second in yellow. Below the queries, there is a toolbar with icons for saving, printing, and running queries. The results of the queries are displayed in a table below the toolbar.

```
SELECT o.Titulo, c.Estado
FROM OfertasDeEmpleo o
JOIN Candidaturas c ON o.IDOferta = c.IDOferta
JOIN Aspirantes a ON c.IDAspirante = a.IDAspirante
WHERE a.IDAspirante = 7;

SELECT *
FROM OfertasDeEmpleo
WHERE Ciudad = 'Ciudad B';
```

Salida de Script x | Resultado de la Consulta x | Resultado de la Consulta 1 x | Resultado de la Consulta 2 x

Todas las Filas Recuperadas: 2 en 0,003 segundos

	TITULO	ESTADO
1	Asistente Administrativo	Finalista
2	Diseñador Gráfico	Pendiente

10. Documentación del diseño (1 punto): Documenta el diseño de la base de datos, incluyendo las tablas, los atributos, las relaciones, las restricciones y las claves primarias. Asegúrate de que la documentación sea clara y comprensible.