# Prueba Técnica

Guía de Instalación, Configuración y Uso

## Levantamiento de la Base de Datos

Para levantar la base de datos, asegúrate de tener Docker instalado en tu máquina. Luego, ejecuta el siguiente comando en la terminal:

**docker-compose up -d**

Esto iniciará un contenedor con Oracle y creará la base de datos necesaria para la aplicación.

## Scripts

Dentro del archivo `scripts.sql` se encuentra un script que obtiene la productividad por ejecutivo.

## Configuración del Proyecto de NestJS

Crea un archivo `.env` en la raíz del proyecto con el siguiente contenido:

ORACLE\_HOST=localhost  
ORACLE\_PORT=1521  
ORACLE\_USER=crm\_user  
ORACLE\_PASSWORD=crm123  
ORACLE\_SERVICE\_NAME=XEPDB1

## Ejecución de la Aplicación

### Requisitos Previos

Asegúrate de tener Node.js, npm y NestJS CLI instalados en tu máquina.

### Instalación de Dependencias

**npm install**

### Inicio de la Aplicación

**npm run start:dev**

La aplicación estará disponible en http://localhost:3000

## Endpoints Disponibles

• GET /api/porductividad/all — Productividad de todos los Ejecutivos.

• GET /api/porductividad/:ejecutivo — Productividad de un Ejecutivo específico.

## Modo de Uso del SSIS

Abrir el proyecto y cargar "carga ventas.siss". Usar las credenciales:

• Usuario: crm\_user

• Contraseña: crm123

Ajustar la ruta del archivo CSV y ejecutar el proceso.

### Funcionamiento del SSIS

1. Limpieza de VENTAS\_STAGING para evitar duplicados.

2. Lectura del archivo CSV desde la ruta configurada.

3. Validación de datos mediante Script Component.

4. Inserción en VENTAS\_STAGING y registro de errores en VENTAS\_ERRORES.

5. Traspaso de datos válidos a la tabla final VENTAS.