

# Modelo de Datos para Practicar Hints en Oracle 11g XE

Este documento describe un modelo de datos simple que permite poner en práctica los conceptos de 'hints' en Oracle 11g XE. Utilizaremos dos tablas, 'empleados' y 'departamentos', para ejecutar consultas y forzar ciertos comportamientos del optimizador de Oracle utilizando hints.

## Descripción del Modelo de Datos

El modelo de datos consta de dos tablas:

- **\*\*departamentos\*\***: Contiene los departamentos de la empresa.
  - dept\_id: Identificador único del departamento (PK).
  - nombre\_departamento: Nombre del departamento.
- **\*\*empleados\*\***: Contiene los empleados y el departamento al que pertenecen.
  - emp\_id: Identificador único del empleado (PK).
  - nombre: Nombre del empleado.
  - dept\_id: Identificador del departamento (FK).

## Paso 1: Crear las Tablas

A continuación se muestra el script para crear las tablas `departamentos` y `empleados`:

```
```sql
```

```
CREATE TABLE departamentos (  
    dept_id NUMBER PRIMARY KEY,  
    nombre_departamento VARCHAR2(100)  
);
```

```
CREATE TABLE empleados (  

```

```
emp_id NUMBER PRIMARY KEY,  
  
nombre VARCHAR2(100),  
  
dept_id NUMBER,  
  
FOREIGN KEY (dept_id) REFERENCES departamentos(dept_id)  
  
);  
...
```

## Paso 2: Insertar Datos

Script para insertar datos en las tablas `departamentos` y `empleados`:

```
```sql  
  
-- Insertar datos en 'departamentos'  
  
INSERT INTO departamentos (dept_id, nombre_departamento)  
  
VALUES (10, 'Ventas');  
  
INSERT INTO departamentos (dept_id, nombre_departamento)  
  
VALUES (20, 'Marketing');  
  
INSERT INTO departamentos (dept_id, nombre_departamento)  
  
VALUES (30, 'Recursos Humanos');  
  
  
-- Insertar datos en 'empleados'  
  
INSERT INTO empleados (emp_id, nombre, dept_id)  
  
VALUES (1, 'Juan', 10);  
  
INSERT INTO empleados (emp_id, nombre, dept_id)  
  
VALUES (2, 'Ana', 20);  
  
INSERT INTO empleados (emp_id, nombre, dept_id)  
  
VALUES (3, 'Pedro', 10);  
  
INSERT INTO empleados (emp_id, nombre, dept_id)
```

```
VALUES (4, 'Luis', 30);
```

```
INSERT INTO empleados (emp_id, nombre, dept_id)
```

```
VALUES (5, 'Maria', 30);
```

```
---
```

### Paso 3: Crear Índices

Crear un índice en la columna `dept\_id` de la tabla `empleados` para optimizar las consultas:

```
```sql
```

```
CREATE INDEX idx_emp_dept ON empleados(dept_id);
```

```
---
```

### Paso 4: Consultas con Hints

A continuación se muestran ejemplos de consultas utilizando hints en Oracle 11g XE.

1. **\*\*FULL\*\***: Forzar una exploración completa de la tabla `empleados`:

```
```sql
```

```
SELECT /*+ FULL(emp) */ emp.nombre, dept.nombre_departamento
```

```
FROM empleados emp, departamentos dept
```

```
WHERE emp.dept_id = dept.dept_id;
```

```
---
```

2. **\*\*INDEX\*\***: Forzar el uso del índice en la columna `dept\_id`:

```
```sql
```

```
SELECT /*+ INDEX(emp idx_emp_dept) */ emp.nombre, dept.nombre_departamento
```

```
FROM empleados emp, departamentos dept
```

```
WHERE emp.dept_id = dept.dept_id;
```

```
...
```

3. **USE\_NL**: Forzar el uso de un Nested Loops Join:

```
```sql
```

```
SELECT /*+ USE_NL(emp dept) */ emp.nombre, dept.nombre_departamento
```

```
FROM empleados emp, departamentos dept
```

```
WHERE emp.dept_id = dept.dept_id;
```

```
...
```

4. **LEADING**: Especificar el orden de escaneo de las tablas en el join:

```
```sql
```

```
SELECT /*+ LEADING(emp) */ emp.nombre, dept.nombre_departamento
```

```
FROM empleados emp, departamentos dept
```

```
WHERE emp.dept_id = dept.dept_id;
```

```
...
```

## Paso 5: Consultar el Plan de Ejecución

Para verificar cómo se está ejecutando una consulta, puedes usar `EXPLAIN PLAN` para ver el plan de ejecución:

```
```sql
```

```
EXPLAIN PLAN FOR
```

```
SELECT /*+ FULL(emp) */ emp.nombre, dept.nombre_departamento
```

```
FROM empleados emp, departamentos dept
```

```
WHERE emp.dept_id = dept.dept_id;
```

```
-- Mostrar el plan de ejecución
```

```
SELECT * FROM TABLE(DBMS_XPLAN.DISPLAY);
```

```
```
```

## **Conclusión**

Este modelo de datos y los ejemplos de consultas te permitirán practicar el uso de hints en Oracle 11g XE. Es importante usar los hints con precaución y probar el rendimiento de las consultas para asegurarse de que el plan de ejecución es el más eficiente posible.