


Se lanza el script SQL con el siguiente comando.

```
EXPLAIN PLAN FOR
SELECT * from employees empl where empl.phone_number='515.123.4568';
select * from table(dbms_xplan.display);
```

obteniéndose el siguiente EXPLAIN sin el uso de índices.

 SQL | All Rows Fetched: 13 in 0.039 seconds


PLAN_TABLE_OUTPUT							
Plan hash value: 1445457117							
-----							
Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time	
-----							
0	SELECT STATEMENT		1	69	3 (0)	00:00:01	
* 1	TABLE ACCESS FULL	EMPLOYEES	1	69	3 (0)	00:00:01	
-----							
Predicate Information (identified by operation id):							
-----							
1 - filter("EMPL"."PHONE_NUMBER"='515.123.4568')							

Como se puede observar en la línea, la operación es un acceso a la tabla FULL, lo que nos deja entender que no hace uso de índices para localiza los números de teléfono.

Después creo un índice para los números de teléfono del siguiente modo.

```
create INDEX EMP_CONTACT_IX
ON EMPLOYEES(PHONE_NUMBER);
```

Tras crear el índice, vuelvo a lanzar el mismo comando con EXPLAIN.

 SQL | All Rows Fetched: 14 in 0.021 seconds

PLAN_TABLE_OUTPUT							
Plan hash value: 2775623239							
-----							
Id	Operation	Name	Rows	Bytes	Cost (%CPU)	Time	
-----							
0	SELECT STATEMENT		1	69	2 (0)	00:00:01	
1	TABLE ACCESS BY INDEX ROWID BATCHED	EMPLOYEES	1	69	2 (0)	00:00:01	
* 2	INDEX RANGE SCAN	EMP_CONTACT_IX	1		1 (0)	00:00:01	
-----							
Predicate Information (identified by operation id):							
-----							
2 - access("EMPL"."PHONE_NUMBER"='515.123.4568')							

Como se puede ver se hace un acceso a la tabla, pero hace uso del índice creado para hacer el scan, lo que le facilita la búsqueda de los valores y agiliza la consulta.