

Contenido

1Ejercicios de Optimización de consultas sobre la base de datos Jardineria.....3

1 Ejercicios de Optimización de consultas sobre la base de datos Jardineria.

1. Consulte cuáles son los índices que hay en la tabla producto utilizando las dos instrucciones SQL que nos permiten obtener esta información de la tabla.

1 SHOW index FROM product

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type
producto	0	PRIMARY	1	codigo_producto	A	276	(NULL)	(NULL)		BTREE
producto	1	gama	1	gama	A	8	(NULL)	(NULL)		BTREE
producto	1	idx_proveedor	1	proveedor	A	16	(NULL)	(NULL)	YES	BTREE
producto	1	idx_nombre_descripcion	1	nombre	(NULL)	(NULL)	(NULL)	(NULL)		FULLTEXT
producto	1	idx_nombre_descripcion	2	descripcion	(NULL)	(NULL)	(NULL)	(NULL)	YES	FULLTEXT

2. Haga uso de EXPLAIN para obtener información sobre cómo se están realizando las consultas y diga cuál de las dos consultas realizará menos comparaciones para encontrar el producto que estamos buscando. ¿Cuántas comparaciones se realizan en cada caso? ¿Por qué?.

SELECT *

FROM producto

WHERE codigo_producto = 'OR-114';

Host: 127.0.0.1 Base de datos: jardineria Consulta

```

1 EXPLAIN
2 SELECT *
3 FROM producto
4 WHERE codigo_producto = 'OR-114';
5

```

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	producto	const	PRIMARY	PRIMARY	62	const	1	Extra

```
SELECT *
FROM producto
WHERE nombre = 'Evonimus Pulchellus';
```

Host: 127.0.0.1 Base de datos: jardineria Consulta*

```

1  EXPLAIN
2  SELECT *
3  FROM producto
4  WHERE nombre = 'Evonimus Pulchellus';

```

producto (1r x 10c)									
id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	producto	ALL	idx_nombre_descripcion	(NULL)	(NULL)	(NULL)	276	Using where

3. Suponga que estamos trabajando con la base de datos jardineria y queremos saber optimizar las siguientes consultas. ¿Cuál de las dos sería más eficiente?. Se recomienda hacer uso de EXPLAIN para obtener información sobre cómo se están realizando las consultas.

```
SELECT AVG(total)
FROM pago
WHERE YEAR(fecha_pago) = 2008;
```

```

1  EXPLAIN
2  SELECT AVG(total)
3  FROM pago
4  WHERE YEAR(fecha_pago) = 2008;

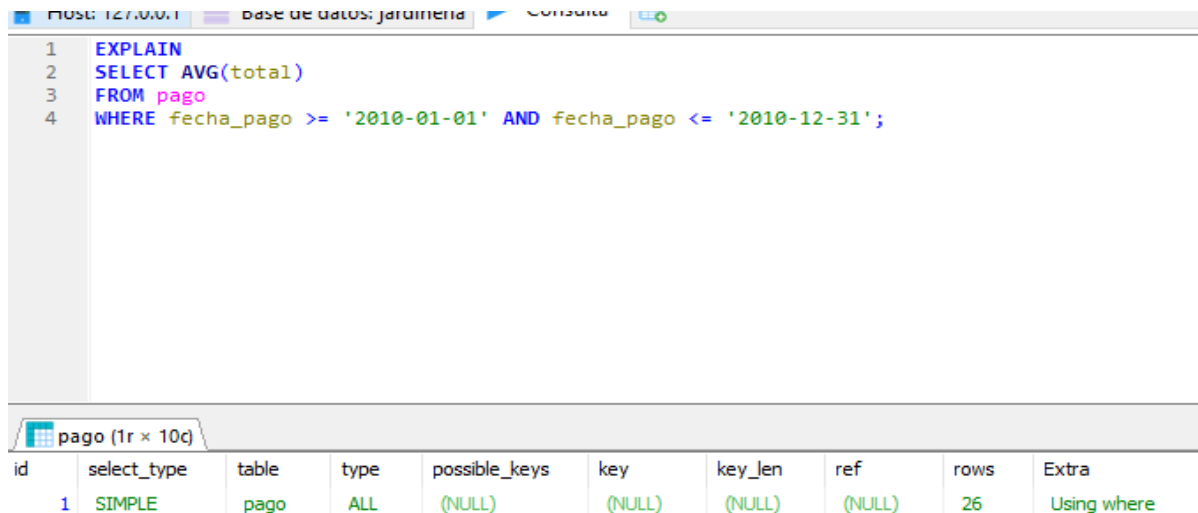
```

pago (1r x 10c)									
id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	pago	ALL	(NULL)	(NULL)	(NULL)	(NULL)	26	Using where

```

SELECT AVG(total)
FROM pago
WHERE fecha_pago >= '2010-01-01' AND fecha_pago <= '2010-12-31';

```



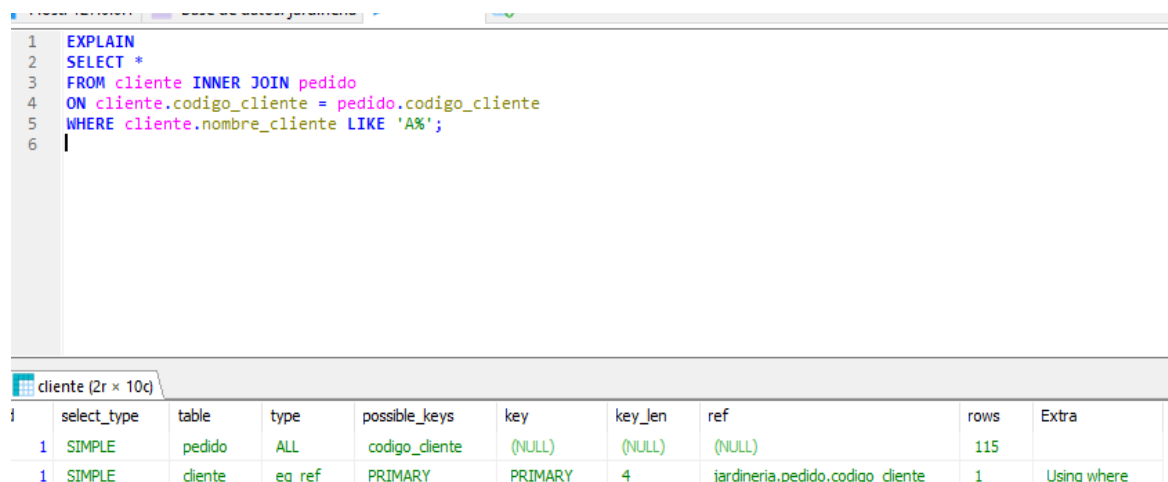
id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	pago	ALL	(NULL)	(NULL)	(NULL)	(NULL)	26	Using where

4. Optimiza la siguiente consultas creando índices cuando sea necesario. Se recomienda hacer uso de EXPLAIN para obtener información sobre cómo se están realizando las consultas.

```

SELECT *
FROM cliente INNER JOIN pedido
ON cliente.codigo_cliente = pedido.codigo_cliente
WHERE cliente.nombre_cliente LIKE 'A%';

```



id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	pedido	ALL	codigo_cliente	(NULL)	(NULL)	(NULL)	115	
1	SIMPLE	cliente	eq_ref	PRIMARY	PRIMARY	4	jardineria.pedido.codigo_cliente	1	Using where

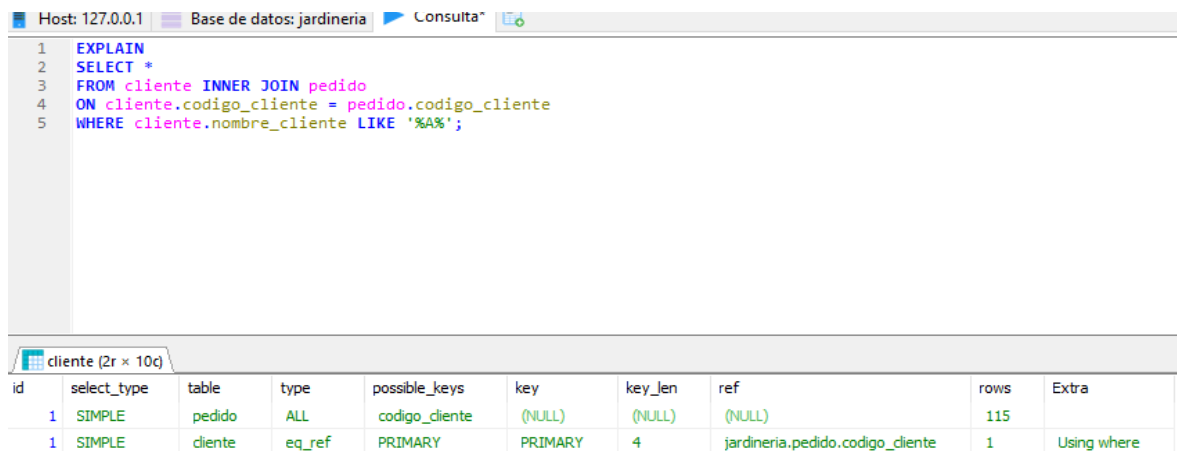
5. ¿Por qué no es posible optimizar el tiempo de ejecución de las siguientes consultas, incluso haciendo uso de índices? Prueba a optimizar dichas consultas con el índice adecuado.

SELECT *

FROM cliente INNER JOIN pedido

ON cliente.codigo_cliente = pedido.codigo_cliente

WHERE cliente.nombre_cliente LIKE '%A%';



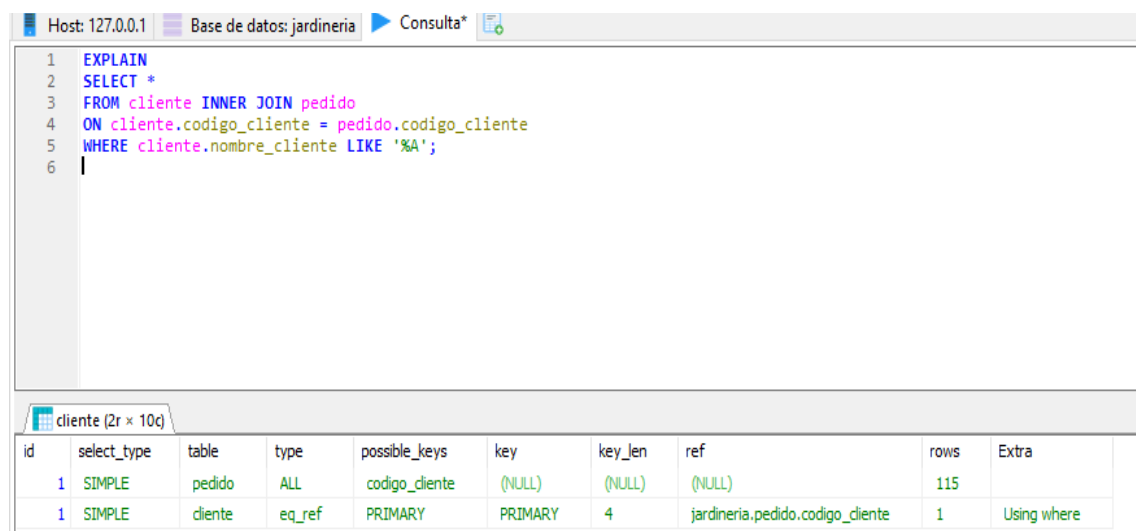
id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	pedido	ALL	codigo_cliente	(NULL)	(NULL)	(NULL)	115	
1	SIMPLE	cliente	eq_ref	PRIMARY	PRIMARY	4	jardineria.pedido.codigo_cliente	1	Using where

SELECT *

FROM cliente INNER JOIN pedido

ON cliente.codigo_cliente = pedido.codigo_cliente

WHERE cliente.nombre_cliente LIKE '%A%';



id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	pedido	ALL	codigo_cliente	(NULL)	(NULL)	(NULL)	115	
1	SIMPLE	cliente	eq_ref	PRIMARY	PRIMARY	4	jardineria.pedido.codigo_cliente	1	Using where

