

## Laboratório. Exercícios.

### Índices

1. Imagine a seguinte situação: o cliente (um grande varejista que atua em toda região sudeste) solicita que construamos um relatório que apresente as vendas por estado, por mês/ano desejado. Ou seja, devemos construir um select como abaixo:  
`SELECT SUM(valor_vendas), estado from VENDAS WHERE mesano = <var> group by estado;`  
Onde mesano é um valor no formato YYYYMM.  
Qual tipo de índice mais indicado para atender a consulta, justifique.
2. Imagine a seguinte situação: o cliente (um grande varejista que atua em toda região sudeste) solicita que construamos a tabela de vendas, onde a mesma terá o campo NotaFiscal como chave primária. Todas as consultas serão realizadas por este campo (NotaFiscal). Informe se é necessário ou não criar índice para este campo e explique o motivo.
3. Imagine que uma Universidade vai construir um sistema. A tabela aluno possuirá o campo CPF e tal campo será a chave primária de tal tabela. Além disso para cada aluno que se inscrever na universidade será gerado um número de matrícula único e sequencial. Deseja-se efetuar consultas pelo campo matricula. Qual o tipo de índice mais apropriado para este caso?
4. Imagine que uma Universidade vai construir um sistema. A tabela nota possuirá o RA do aluno, AV1 e AV2. Deseja-se fazer consultas baseados na média do aluno onde a média é  $(AV1+AV2)/2$ . Tipos de consulta que se deseja:  
`SELECT RA from nota where (AV1+AV2)/2 >=6;`  
Qual o tipo de índice mais apropriado para este caso?
5. Rode os seguintes comandos (sql developer como usuário system). É um processo demorado pois será criada uma tabela com 10 milhões de linhas.

`set timing on;`

`create tablespace teste datafile 'C:\teste.dbf'  
size 100M autoextend on NEXT 100m;`

`drop table teste;  
create table teste (cod number(13),nome varchar(50)) tablespace teste;`

```
begin
  for i in 1 .. 10000000
  loop
    insert into teste values (i,'Leonardo#' || to_Char(i));
    commit;
  end loop;
end;
```

Em seguida verifique o tempo de execução necessário para a execução do comando abaixo (clique no segundo botão para ver o tempo de execução):  
`select * from teste where cod=3432326;`

Em seguida crie um índice (árvore B) para esta mesma tabela de forma a deixar a consulta eficiente e repita a consulta e verifique o tempo de execução necessários.

6. Dada as consultas abaixo crie a menor quantidade de índices que aumentarão o desempenho das mesmas (considere que todas tabelas são grandes).

Situação 1

```
DROP TABLE FUNCIONARIO CASCADE CONSTRAINT;
CREATE TABLE FUNCIONARIO (MATRICULA NUMBER(10) PRIMARY KEY, CPF
VARCHAR(11) UNIQUE, CPFRESP VARCHAR(11), NOME VARCHAR(100) );
SELECT * FROM FUNCIONARIO WHERE MATRICULA = <VAR>;
SELECT * FROM FUNCIONARIO WHERE CPF = <VAR>;
SELECT * FROM FUNCIONARIO WHERE CPFRESP = <VAR>;
```

Situação 2 – Varejista de grande porte.

```
DROP TABLE CLIENTE;
DROP TABLE PRODUTO;
DROP TABLE VENDEDOR;
CREATE TABLE VENDEDOR (MAT NUMBER(10) PRIMARY KEY, NOME
VARCHAR(30));
CREATE TABLE PRODUTO (COD NUMBER(10) PRIMARY KEY, DESCRICAO
VARCHAR(30));
CREATE TABLE CLIENTE (COD NUMBER(10) PRIMARY KEY, DOC NUMBER(14),
TPPESSOA NUMBER(1), NOME VARCHAR(30));
SELECT * FROM VENDEDOR WHERE MAT = <VAR>;
SELECT * FROM VENDEDOR WHERE NOME = <VAR>;
SELECT * FROM PRODUTO WHERE COD = <VAR>;
SELECT * FROM PRODUTO WHERE NOME LIKE '%<VAR>%';
SELECT * FROM CLIENTE WHERE TPPESSOA = <VAR> AND DOC=<VAR>;
SELECT * FROM CLIENTE WHERE NOME LIKE '<VAR>%';
```

Situação 3 – Considere a situação 2 com a inclusão da tabela vendas abaixo e a consulta a seguir

```
CREATE TABLE VENDAS (MAT NUMBER(10), CODPROD NUMBER(10), CODCLI
NUMBER(10), VALOR NUMBER(10,2), QTDE NUMBER(2),
CONSTRAINT VD1_FK FOREIGN KEY (MAT) REFERENCES VENDEDOR(MAT),
CONSTRAINT VD2_FK FOREIGN KEY (CODPROD) REFERENCES PRODUTO(COD),
CONSTRAINT VD3_FK FOREIGN KEY (CODCLI) REFERENCES CLIENTE(COD)).
```

```
SELECT * FROM VENDAS V
INNER JOIN CLIENTE C ON V.CODCLI=C.COD
INNER JOIN PRODUTO P ON V.CODPROD=P.COD
INNER JOIN VENDEDOR E ON V.MAT=E.MAT
WHERE V.CODCLI = <var>;
```

```
SELECT * FROM VENDAS V
INNER JOIN CLIENTE C ON V.CODCLI=C.COD
INNER JOIN PRODUTO P ON V.CODPROD=P.COD
INNER JOIN VENDEDOR E ON V.MAT=E.MAT
WHERE V.CODPROD = <var>;
```

```
SELECT * FROM VENDAS V
INNER JOIN CLIENTE C ON V.CODCLI=C.COD
INNER JOIN PRODUTO P ON V.CODPROD=P.COD
INNER JOIN VENDEDOR E ON V.MAT=E.MAT
WHERE V.MAT = <var>;
```