

1) Descreva o funcionamento das chaves estrangeiras. Explique se são utilizados índices para manutenção da integridade.

Chave estrangeira ou 'foreign key' é um campo que relaciona duas tabelas, como padrão, uma chave estrangeira referencia os atributos de chave primária da tabela referenciada, listando os atributos que compõem a chave estrangeira e o nome da relação referenciada pela chave estrangeira, garantindo a integridade referencial, ou seja, garantindo que um valor que aparece em uma relação para um determinado conjunto de atributos também apareça para um certo conjunto de atributos em outra relação.

Índice é uma estrutura (ou arquivo) auxiliar associado a um conjunto de dados (ou tabela), é referência associada a uma chave. Sua função é acelerar o tempo de acesso às linhas de uma tabela, criando ponteiros para os dados armazenados em colunas específicas. Como integridade de dados em um banco de dados é a manutenção, precisão e consistência de dados, o índice de certa forma garante integridade, pois melhora a precisão de dados que um banco de dados tem, mesmo que seja usado fins de otimização.

2) Explique os métodos convencionais para organização de registros de tamanho fixo e de tamanho variável em arquivos. Comente sobre o acesso direto a registros pela posição tanto em registros de tamanho fixo quanto variável.

Registro é conjunto de atributos que pertencem a um mesmo elemento quando o arquivo é visto em termos de uma organização de alto nível. Em registro de tamanho fixo, já é alocado um espaço definido para todos os registros, com isso, registros podem ter espaços em branco (espaços não usuais) que ocupam memória, ou a informação pode ter que usar dois ou mais espaços de registro, mas as operações de inserção, busca e remoção são bem mais simples, visto que basta somente encontrar a posição desejável para tal.

Já registro de tamanho variável ocupam menos da memória, pois não existe espaços em branco, sendo todos os registros colocados justapostos, utilizando outras estruturas para representar o início ou fim dos registros. Um exemplo pode ser iniciar cada registro com um indicador de tamanho consistindo da contagem do número de bytes do registro (o mais comum), ou pode-se usar um segundo arquivo para armazenar o endereço do byte de início de cada registro, dentre outras formas. Inserção busca e remoção em registro de tamanho variável é bem mais complexo, pois precisam saber a posição inicial e final do registro para realizar tal operação.

Para ter o acesso direto aos registros pela posição, em registros de tamanho fixo, basta ir à posição desejada, visto que cada registro tem o mesmo tamanho, basta fazer:

$\text{posição} \times \text{tamanho dos registros} = \text{posição do registro},$

com isso obtemos a posição do registro que procuramos para o acesso.

Já registros de tamanho variável, é necessário fazer quase uma busca sequencial, pois temos que ver o tamanho do registro atual para pular para o próximo registro, fazendo essa verificação até encontrar a posição desejada. Por exemplo:

Posição: 3

Ler o tamanho do registro de primeira posição, pular o tamanho encontrado, ler o tamanho do registro de segunda posição, pular o tamanho encontrado, acesso ao registro 3.

3) Descreva o funcionamento da estrutura denominada lista invertida. Apresente o desenho da representação do índice secundário e da lista invertida após inserção dos seguintes pares <chave,valor>

Listas invertidas são uma estrutura de dados que mapeia termos às suas ocorrências em um documento ou conjunto de documentos, sendo que uma chave secundária leva a um conjunto de uma ou mais chaves primárias. A ideia de listas invertidas mais otimizada utiliza listas ligadas, em que é usado um índice secundário com um atributo contendo o número relativo da primeira ocorrência da chave no arquivo da lista invertida.

Índice secundário	
Chave secundaria	Lista
ANA	15
BEATRIZ	20
CECILIA	2
JULIA	18
MARIA	16
VILMA	19

Lista invertida		
Posição	Chave primaria	Ponteiro
1	1	-1
2	32	-1
3	57	-1
4	107	-1
5	149	-1
6	193	1
7	240	6
8	286	-1
9	330	3
10	357	9
11	404	4
12	434	11
13	475	12
14	513	10
15	562	8
16	598	7
17	637	5
18	669	13
19	712	14
20	745	17