**Sistema** – é um conjunto de componentes dinamicamente interrelacionados e interdependentes que interajam no desempenho de uma actividade ou função com o proposito de atingirem objectivos comuns. É constituído por subsistemas relacionados entre si, ocorrendo comunicação entre eles. É composto por quatro funções basicas: input, tratamento, armazenamento e output.

**Informação** – é uma colecção de factos de onde se podem extrair conclusões;Objecto formatado (dotado de formas reconhecíveis), criado artificialmente pelo homem a fim de representar um acontecimento percebido por ele no mundo real.

A informação na organização flui de 2 formas, entre os niveis de gestão e dentro de cada nivel de gestão.

**Tipos de informação nas Organizações**

**Operacional** – É utilizada diariamente e permite que a organização leve a cabo as suas actividades de rotina, de forma eficiente. Propriedades: Raio de acção reduzido, muito detalhada, em bruto, actual, exacta, quantitativa e interna.

**Gestão** – Serve de suporte à tomada de decisões nos 3 níveis de gestão (Estratégico, táctico e operacional) Propriedades: Raio de acção amplo, pouco detalhada, sumariada, não precisa ser actual, não precisa ser exacta, qualitativa e externa.

**Sistema de informação** – é uma infraestrutura automatizada ou manual, que agindo de forma integrada é capaz de dar apoio aos processos de recolha, armazenamento, criação e distribuição de informações certas, para as pessoas certas, no momento certo, na quantidade certa e no formato certo.

**Planeamento** - é a actividade de identificação de políticas, definição de objectivos e construçao de planos e orçamentos em que sejam contemplados os objectivos de gestão da organização e do sistema de informação. Deve incluir:

* Descrição da situação actual, incluindo uma análise de pontos fortes e fracos, análise dos níveis de serviço e dos objectivos da organização;
* Conjunto dos modelos que representam a arquitectura da informação, aplicacional e tecnológica;
* Proposta de projectos a desenvolver e respectivo calendário;
* Avaliação dos recursos e dos meios necessários para a implementação dos projectos;
* Plano de acompanhamento.

**Desenvolvimento e sistemas de informação** – é o processo de mudança que visa melhorar o desempenho de um subsistema de informação, abrangendo as fases:

* Estudo preliminar (definição dos objectivos),
* Identificação de requisitos (indicar os requisitos para o SI, analisar tipos de utilizadores - internos, externos), análise detalhada (org. de regras e procedimentos),
* Desenho (com especificações técnicas)
* Codificação, (através de linguagens de programação ou de SGBD);
* Testes de implementação (realização de testes para garantir a satisfação dos objectivos);
* Manutenção (correctiva-erros de sistema, perfectiva-alterar a funcionalidade do sistema).

**Sistema de informação organizacional -** é composto por recursos computacionais humanos e organizacionais que agindo de forma integrada, apoiam os processos de recolha, armazenamento, criação e distribuição da informação. A importância que assume na perspectiva da gestão de bibliotecas prende-se com a garantia de integração e centralização do grande fluxo de informação existente e o de permitir criar processos de cruzamento de dados que vão garantir a eficiência das pesquisas necessárias ao seu bom funcionamento. Tem por objectivo satisfazer adequadamente a totalidade das necessidades de informação de uma organização e dos respectivos processos de negócio. Quando este conceito surge no âmbito das organizações, está usualmente relacionado com os conceitos de mudança e a capacidade adaptativa da organização a esta mudança, apresentando o conceito dinâmico de organização, cultura e mudança organizacional. Visa a mudança das organizações de modo a que se transformem em sistemas sociais que incorporam a mudança como componente intrínseca.

**Fases de introdução de SI nas organizações**

Estádios de crescimento de Nolan – I – Iniciação; II – Expansão; III – Controlo; IV – Integração; V – Administração de dados; VI – Maturidade.

* Iniciação - instalação do pc e automatização
* Expansão - rápida proliferação e crescimento dos sistemas informaticos
* Controlo - crescimento, aumento de custos, necessidade de um controlo
* Integração - necessidade de integrar diversas aplicações em sistema coerente
* Administração dos dados - gestão da informação é feita a conclusão do desenvolvimento e implementação de um sistema integrado de bases de dados
* Maturidade - o sistema está integrado no sistema produtivo da empresa.

**Sistema de base de dados**- sistema baseado em computadores cujo objectivo é armazenar dados que permitam aos utilizadores acederem á informação solicitada. Contem um repositório de dados que é acedido por diversos utilizadores através de um sistema SGBD (sist gestão base dados) com as seguintes características:

**Catálogo do sistema** – é a definição completa ou descrição da base de dados. A informação contida no catálogo é chamada de metadados e está acessível ao SGBD e aos utilizadores, sendo criado automaticamente pelo sistema.

**Interdependência Dados/Aplicações** – A estrutura de dados está gravada no catálogo do SGBD e separada das aplicações que lhe acedem.

**Abstracção dos dados** – O SGBD deve esconder os detalhes sobre a forma como os dados são fisicamente armazenados e mantidos.

**Vistas múltiplas de utilizadores** – Cada utilizador necessita de ter acesso a perspectivas diferentes da base de dados. Uma vista pode ser um subconjunto da base de dados. Na vista incluem-se também as permissões relativas ao tipo de acessos que cada utilizador pode ter (leitura, escrita, ambos).

**Intervenientes no SGD**

* Adm. de dados e administradores
* Analistas base de dados
* Programadores de aplicações
* Utilizadores finais;

**Funções do SGBD**

* **Gestão do armazenamento dos dados** cria estruturas necessarias ao armazenamento e utilização de dados, libertando o utilizador da dificil tarefa de definir e programar as suas características físicas. É criado o dicionário de dados que facilita o processo;
* **Gestão de interface** formata os dados consultados de modo a obter um aspecto perceptivel pelo utilizador. São disponibilizadas linguagens de programação que permitem essa interface. Fornecem capacidades para comunicar em ambientes de rede de computadores através da Net;
* **Gestão de segurança** protege a BD de acessos não autorizados. São impostas regras que definem quais os utilizadores que podem ter acesso à BD e para os utilizadores autorizados a forma como pode ser feito esse acesso, quais os ficheiros que podem ser acedidos.
* **Manutenção de integridade de dados** (assegura a verificação de restrição de integridade de modo que a BD contenha sempre dados validos minimizando a redundancia nos dados e maximizando a sua consistência. Devem fazer cópias de segurança assegurando a integridade dos dados;
* **Controlo de acessos por vários utilizadores em simultâneo** deve fornecer mecanismos que assegurem que a BD é atualizada quando varios utilizadores acedem aos mesmos dados em simultâneo (controlo da concorrência);

**Objetivos da SBD** fornecer aos utilizadores uma visão abstrata dos dados escondendo detalhes sobre a forma como os dados são armazenados e mantidos fisicamente. Feita através da arquitetura ANSI/SPARC composta por 3 niveis- interno, conceptual, externo.

**Niveis de abstracção de um sistema de base de dados** pertencem à arquitectura ANSI/SPARS:

* Nivel interno- descreve a forma como os dados são fisicamente armazenados de forma pormenizada e consiste no nivel mais baixo de abstracção. [**SDL**](#SDL) – linguagens de definição do armazenamento, utilizada para especificar o esquema interno
* Nível conceptual – descreve a estrutura dos dados para uma comunidade de utilizadores, quais os dados que estão armazenados e quais as relações existentes entre esses dados. Escondem os detalhes das estruturas físicas
* Nível externo – Nível mais elevado de abstracção e inclui vistas externas ou vistas do utilizador que descrevem apenas uma parte da base de dados. [VDL](#VDL)- linguagem de definição de vistas, utilizada para especificar as vistas de cada um dos utilizadores.

Esta arquitectura pressupõe ainda a correspondência entre os níveis conceptuais e externo e niveis conceptual e interno.

**Interdependência de dados** - é a capacidade de alterar o esquema conceptual de um determinado nível sem alterar o esquema do proximo nivel.

Existem 2 tipos de interdependência:

**Independência lógica -** a capacidade de alterar o esquema conceptual sem ter que alterar os esquemas externos ou aplicações. Pode-se alterar o esquema para expandir a base dados, para alterar restrições ou reduzir a base dados

**Independência física** - capacidade de alterar esquema interno sem alterar o esquema externo. As alterações internas podem ser feitas para reorganizar fisicamente os ficheiros para melhorar o desempenho do sistema.

**Linguagens de SGBD**

* DDL – (Data Definition Language) linguagem de definição de dados, utlizada para definir um esquema ou modificar o esquema existente
* DML – (Data Manipulation L) linguagem para manipulação de dados, utilizada para efectuar operações basicas de manipulação de dados (inserir, modificar, armazenar ou apagar dados)
* SDL – (Storage Definition) linguagens de definição do armazenamento, utilizada para especificar o esquema interno
* VDL- (View Definition) linguagem de definição de vistas, utilizada para especificar as vistas de cada um dos utilizadores
* SQL – (Structured Query Language) a mais usada **SQL** é uma linguagem padrão no modelo relacional para a criação da estrutura de bases de dados, os seus acessos, as suas consultas e actualizações da informação. O SQL comunica com o SGBD aquilo que pretende da base de dados. Dividem-se em sub-linguagens; **DDL** e o **DML.** Disponibiliza um conjunto de comandos para Criar, alterar e apagar Tabelas e outras estruturas

**Modelo de Bases de Dados** – Colecção de estruturas lógicas para representar estrutura dos dados e as relações existentes entre eles na BD.

**Modelo Conceptual (E\_R**) – Utilizado para se obter a descrição lógica do sistema e preocupa-se com o que está na BD.

**Vantagens** – Diminuição da redundância, Aumenta a integridade, diminui o espaço ocupado no disco, Partilha de dados, controlo centralizado de dados, facilidade de mudança, utilização linguagens de interrogação, isolamento entre os programas e as modificações à BD, maior facilidade manutenção.

**Desvantagens** – Necessidade de pessoal especializado, Interferência com dados partilhados, conflito organizacional.

**Relação** – Estrutura lógica representada por 1 tabela com colunas e linhas.

**Atributo** – Coluna de uma relação à qual atribuímos um nome.

**Entidade** – Conjunto de pessoas, objectos ou conceitos sobre os quais queremos guardar dados.

**Tuplo** – Linha de uma relação

**Grau de uma relação** – Número de atributos de uma relação.

**Cardinalidade** – Numero de linhas de uma relação.

**Domínio** – Conjunto de valores permitidos para um ou mais atributos.

**Chave estrangeira** – Atributo ou conjunto de atributos de uma relação que existem também como chave primária noutra relação.

**Comandos SQL**

**Create** – Para definir a estrutura da BD que irá conter as tabelas

**Alter** – Para efectuar alterações na estrutura da tabela de 3 formas: Adicionar, alterar, eliminar.

**Drop** – Permite remover uma BD e registos das tabelas (tuplos)

**Update** – Permite efectuar alterações nos dados já existentes.

**Delete** – Permite eliminar uma ou mais linhas de dados.

**Regras de Codd** (1985)

* Regra inicial
* Representação da informação
* Acesso garantido
* Tratamento de nulos
* Catálogo dinâmico online
* Sublinguagem dados completa
* Actualização das vistas
* Inserção, eliminação e actualização de alto nível
* Independência física de dados
* Independência lógica de dados
* Independência da integridade
* Independência da distribuição
* Regra não subversão.

**Abstracção** – Processo de identificação dos aspectos importantes de um sistema, ignorando os outros.

**Modelo** – É a imagem ou desenho que representa o objecto que se pretende reproduzir. Deve ser simples, não redundante, flexível e adaptável.

**Regras de integridade** – regras que garantem a adequação dos dados

**Integridade de entidade** – Numa relação nenhum dos valores da chave primária pode ser nula.

**Integridade referencial** – Existindo uma chave estrangeira noutra relação, o seu valor tem que ser igual ao da chave primária.

**Modelo E-R** – Modelo conceptual de dados, com um elevado grau de abstracção. Permite que as diferentes perspectivas de dados se aproximem, formando uma estrutura comum.

**Atributos chave** – uma chave é constituída por um atributo ou conjunto de atributos. Uma chave constituída por mais de um atributo diz-se composta

**Super-chave** – atributo ou conjunto de atributos que identificam de forma única um tuplo numa relação.

.Cod-empregado; nome empregado; cod-empregado +nome + morada

**Chave candidata** –Uma super chave pode ter atributos que não são necessários para a identificação dos tuplos de forma única.

Interessa-nos apenas identificar aquelas que contêm apenas os tributos necessários a essa identificação – chave candidata: . cod-empregado; nome empregado

**Chave primária** – De entre as chaves candidatas é escolhida a que identificar de forma única cada tuplo da relação – chave primária: EMPREGADO (cod-empregado, nome empregado, morada do empregado, cidade, telefone, projecto numero.

**Chave estrangeira** – Um atributo ou conjunto de atributos de uma relação que existe também como a chave primária de outra relação – Chave estrangeira

PROJECTO: projecto-numero, desc.proj, local

**Atributo composto** – Pode ser subdividido, gerando novos atributos.

**Atributo simples** – Não pode ser subdividido.

**Atributo derivado** – Calculado a partir de outro atributo.

**Relacionamento** – Associação útil entre duas entidades

**Grau de relacionamento** – Número de entidades participantes num relacionamento (binário, ternário).

**Relacionamento recursivo** – Entidade que participa em mais que um relacionamento mas com papéis diferentes.

**Cardinalidade** – 1:1 – 1:M – M:M

**Participação** – Obrigatória – Opcional.

**Entidade fraca** – Dependência da existência de outra identidade que tenha a sua chave primária.

**Normalização** – Técnica de validação das estruturas através da análise do relacionamento entre atributos.

**Importância da normalização** – Minimizar redundâncias de dados, redução do espaço disco, levantamento das anomalias na inserção, alteração e eliminação de dados.

**Modelo hierárquico**

Foi o primeiro modelo de SGBD.As duas estruturas básicas que se usam são os registos e os relacionamentos pai-filho**.** Os dados são classificados hierarquicamente, em estrutura descendente e usados apontadores entre os diferentes registos.O acesso a qualquer registo é feito a partir da raiz e correndo cada um dos nós da árvore da esquerda para a direita e de cima para baixo.

**Modelo em rede**

Similar ao anterior e utiliza igualmente apontadores para os registos mas a estrutura já não é obrigatoriamente “em árvore” descendente. Neste modelo um relacionamento chama-se set e tem um nome, ao passo que o registo pai se chama owner e o filho member. Facilita a representação de relacionamentos muitos para muitos e o acesso aos dados é mais flexível. No entanto, exige um conhecimento profundo do sistema a desenvolver por parte dos programadores, para que a sua utilização seja eficiente.

**Modelo relacional**

O modelo relacional de bases de dados, apresentado por Edgar F.Codd em 1970, representa uma quebra com os modelos anteriores e uma verdadeira revolução.

* Sistema de gestão de bases de dados relacionais:
* A sua manipulação dá-se de acordo com a teoria matemática da álgebra relacional.
* A estrutura principal é a tabela, sendo esta constituída por linhas (registos) e colunas (atributos ou campos), nos quais os dados são registados, e se relaciona com outras através da partilha de atributos comuns.
* Possui uma linguagem própria de acesso aos dados, denominada SQL (Structured Query Language).

Outras aplicações das bases de dados

No final do séc. XX surgiram novos desenvolvimentos na área das bases de dados: As Data Warehouses; as ferramentas OLAP; as ferramentas Data Mining e a integração das bases de dados com a web.

**Modelo Orientado a Objectos**

A este modelo está associada a noção de reutilização e os dados são armazenados sob a forma de objectos, contendo atributos que os descrevem e métodos que caracterizam o seu comportamento. Os objectos juntam-se em estruturas que chamamos classes de forma a partilharem estruturas e comportamentos. As classes estão organizadas através de uma estrutura hierárquica, na qual os objectos de uma classe herdam os atributos e métodos das suas super-classes.

**Extensões ao modelo relacional**

São um grupo de funcionalidades criadas com o objectivo de fazer face a eventuais fragilidades e permitindo a sobrevivência do modelo relacional em presença de novas e constantes necessidades das BD’s.

* Operadores da álgebra relacional (linguagem ao alto nivel)
* Select, project, product, union, diference, join, intersect, divide
* Dados- conj. de relações ou tabelas
* Relação - estrutura logica representada por uma tabela
* Relações - guardam inf, sobre objetivos que queremos representar na BD
* Atributo - uma coluna de uma relação á qual atribuimos um nome
* Tuplo - linha de uma relação (registo)
* Grau de uma relação - nº de atributos que contem
* Cardinalidade - o nº de tuplos que a relação contém.

**Para que serve a avaliação de um sistema de bibliotecas/sistema de informação?**

A avaliação de uma biblioteca surge quando ocorre algum problema no sistema ou quando é necessario justificar as despesas (despesas orçamentais) com pessoal, materiais ou outros aspectos dos serviços bem como quando há mudanças propostas para os serviços. a avaliação deve envolver a recolha de informação sobre a necessidades e desejos dos utilizadores fazendo corresponder esses ao que a biblioteca oferece.

* **Beneficios**- estão relacionados com as comunidades que servem, isto é, o valor que tem a biblioteca ou serviço para a comunidade (cidade, instituição educacional ou empresa) ou para o utilizador individual podendo na sua base ter a questão do custo.
* **Eficiência** – algo é eficiente quando é om um minimo de recursos para a sua função. Em sistemas de bibliotecas ou informações a eficiencia deve ser medida de acordo com o consumo dos seus recursos- pessoas, materiais e dinheiro. Ex: consulta de catalogo bibliografico online
* **Eficácia** – medida que traduz a aproximação de um sistema às reais necessidades dos utilizadores. Em termos de um sistema de informação de gestão de bibliotecas, a eficacia pode ser medida pelo nº de funcionalidades que um sistema possui. Quanto mais funcionalidades tiver o sistema maior será o nº de utilizadores cujos objectivos são satisfeitos. Por outro lado, há ainda a considerar a rapidez com que essas funcionalidades são executadas, tendo em conta que existindo muitas funcionalidades mas com um percurso muito complexo para se executado levando bastante tempo, a eficacia do sistema baixara drasticamente.
* **Sucesso** – na avaliação de um sistema de informação de gestão de biblioteca o criterio sucesso está associado a novos programas ou outro tipo de inovação e tem subjacente uma percepção subjectiva tendo em conta que o sucesso de uns é o fracassso de outros. Por outro lado o sucesso está dependente dos objectivos do novo programa ou outra inovação.

Ex: qual o objectivo do emprestimo bibliotecario!

* - Atrair novos utilizadores?
* - Colmatar falhas no nosso acervo?

De acordo com os objectivos traçados saber se foram cumpridos- é uma ideia de sucesso

O sucesso é um projecto bem sucedido.

*Comente:*

*Existe uma convergencia crescente entre a informação e a tecnologia em que esta assenta e um desejo de utilizar a tecnologia para suportar o avanço do ensino, da aprendizagem e do proceso de pesquisa e investigação*

Tendo em conta a importancia da informação para os utilizadores e do seu valor para a organização surge como uma necessidade em termos de planeamento, controlo e rotina de trabalho, para que a informação esteja facilmente disponivel, utilizavel, compreensivel e disponivel a tempo de ser obtida a custos razoaveis. Neste contexto a tecnologia é uma mais-valia na concretização destes objectivos, uma vez que permite reunir, guardar, processar e facultar informação relevante.

Em termos de ensino, aprendizagem, pesquisa e investigação as TIC são um contributo essencial, atraves da informação em formato digital, acesso online, transferencias de ficheiros, ligação em rede, partilha de recursos de informação, etc. Com este avanço da informação e as tecnologias estamos á distância de um click em qualquer parte do planeta terra.

**Principais funcionalidades de um sistema de gestão de bibliotecas**

* Sistema de catalogação
* Gestão de utentes, circulação e emprestimos
* Gestão de aquisições
* Gestão de publicações periódicas
* Pesquisa local e Web
* Estatística
* Administração

**Características gerais de um ILS** quando o sistema GB compartilha uma base de dados comuns para executar todas as funções basicas de uma biblioteca o sistema é integrado.

* Módulos funcionais-sistema,
* Sistema operativo,
* Sistema de base de dados,
* Arquitectura de rede-interface de utilizadores,
* Standards de automatização de bibliotecas (MARC e Z39.50)

**RFP**-é um pedido formal para uma oferta por fabricante de soluçoes pa sistemas de automatização de bibliotecas. É um documento que esclarece os contornos do projecto, o ambito, proposito, descrição, requistos minimos para o sistema.

**Automatização de bibliotecas**-são usadas para substituir os sistemas manuais na biblioteca é um software que foi desesnvolvido para lidar com funções de manutenção basica de uma biblioteca. As funções são- aquisição, catalogação, circulação, gestão de periodicos e referencia.

* Modulo de catalogação- usado para a criação, armazenamento, recuperação e gestão de registoa e ou indices. Define o formato do registo usado na Bde disponibiliza o controlo de autor, temas etc.. existem 2 interfaces diferentes, um para os utilizadoresreaslizarem as pesquisas, outro para os catalogadores para manutenção na BD.
* OPAC- oferecido como modelo add-on.o meio de acesso ao catálogo usado pelos utilizadores limita a pesquisa e consulta. Permitem às bibliotecas publicarem os seus catalogos na Web tornando-os acessiveis atraves do web.podem ser ligados ao módulo de circulação para os utilizadores verificarem o estado do livro (empréstimo) para fazer reservas e outras ligações para recursos eletronicos na web em bases de dados.
* Módulo de circulação- emprestimos, devoluções, renovações. Gestao de materiais da biblioteca
* Módulo de aquisição- automatiza o processo de aquisição, usado para estatistica, aquisição online se o sistema estiver ligado a rede externa.
* Módulo de controlo de periodicos- gere colocação, cancelamento, consulte de encomendas, devolução de material etc..poede permitir encomenda de periódicos online.
* Módulo de empréstimos inter-bibliotecas
* Modulo Add-on – oferece funçoes e funcionalidades adicionais e opcionais (relatórios, inventários, exportação de registos de /para formato MARC, Web OPAC Z39.50
* Suporte a standards – facilitam o intercâmbio de dados entre bibliotecas e instituições, pesquisa e informação e reuperação suportados pelos sistemas MARC e Z39.50
* MARC- standards usados para a representação de informação bibliografica e relacionada para livros e outros materiais, numa forma legivel para computadores. Permite ás bibliotecas partilharem recursos bibliograficos com outras bibliotecas.
* Z39.50 – protocolo de pesquisa e recuperação usado primariamente pelos sistemas de suporte a bibliotecas.

**Operadores**

**Operador de álgebra relacional – A operação SELECT é utilizada para** seleccionar determinados atributos numa relação.

**Operador PRODUCT** – Permite combinar informação de várias relações

**Operador UNION** – o resultadoda operação UNION, aplicada a duas relaçõesR e S (RUS) é uma relação que inclui todos os tuplos pertencentesaR ou a S ou ambas as relações.

**Operador DIFFERENCE** – O resultado da operação DIFERRENCE, aplicada a duas relações R e S, é uma relação que inclui todos os tuplos que estão em R mas não em S.

**Operador INTERSET** – O resultado da operação INTERSECT, aplicada a duas relações R e S (R^S) é uma relação que inclui todos os tuplos pertencentes a ambas as relações.

**Operador JOIN** –A operação JOIN permite-nos combinar tuplos relacionados, pertencentes a duas (ou mais) relações.

**Operador DIVISION** – o resultado da operação DIVISION é uma relação queinclui os tuplos R cujos atributos não são atributos de S.

**Queries com SQL**

Uma query é uma questão que o utilizador coloca à base de dados. Todas as questões são baseadas no comando SELECT.

**SELECT** – especifica dos atributos que irão surgir na resposta

**FROM** – especifica a tabela ou tabelas a serem utilizadas

**Where** – filtra as linhas que satisfazem a condição especificada

**Group By** – Forma grupos de linhas com os valores dos atributos indicados iguais, permitindo a utilização de funções de agregação:

**MAX** valor máximo encontrado para aquele atributo,

**MIN** valor mínimo;

**COUNT** número de linhas contendo o atributo especificado;

**SUM** soma de todos os valores do atributo;

**HAVING** – filtra os grupos que satisfazem a condição

**ORDER BY** – especifica quais os atributos que irão funcionar como chave de ordenação.

**Order by** *expressão* ascendente / descendente (default = ascendente [abrv.= asc]. Não necessita ser indicado; descendente é indicado pela abreviatura desc. Ex.: Order by *expressão* desc)

**Notas**: Having utiliza-se por substituição do Where quando se utiliza o Group by

**EXERCÍCIOS**

1. **TABELA LIVRO**

**Cod\_obra data\_aquisição Estante Prateleira Unidade**

**112233 18032007 45 2 4**

a)Escreva o cod SQL permite listar todas as obras adquiridas antes da data 010122008.

SELECT \* FROM Livro

WHERE Data\_Aquisição < ‘010122008’

Datas e texto são referenciados entre pelicas; o valor nulo ou vazio é representado por duas pelicas consecutivas: ‘’

Não se deve utilizar o \* pois vai resultar na listagem de todos os campos. Deve-se listar apenas os campos solicitados, neste exemplo o Cod\_obra

b) Listar o código da obra e a respectiva data de aquisição para todos os livros em que tenham mais de 2 unidades.

SELECT Cod\_obra, Data\_aquisição

FROM Livro

WHERE Unidade > 2

1. **Tabela Empréstimo**

CodUtente CodObra DataEmprestimo DataEntrega

112233 1803 03/04/2011 08/05/2011

1. Escreva o código SQL que permite listar todas as obras já emprestadas ao utente cujo código é 112233

Select CodObra

From Emprestimo

Where CodUtente = 112233

1. Escreva o código SQL que permite listar o Código da Obra para todos os livros emprestados que não tenham sido devolvidos (DataEntrega=’’).

Select CodObra

From Emprestimo

Where DataEntrega = ‘’

1. **Considere a seguinte estrutura de uma tabela relacional:**

**Tabela Livro**

Cod\_Obra Título Ano\_Impressao Ano\_Aquisicao Cod\_Tipo

12334 XPTO 2003 2003 4

**Tabela Tipo**

Cod\_Tipo Descricao

4 Ficcao

**a)** Escreva o código SQL que permite listar o título de todas as obras adquiridas depois do ano 2000.

Select titulo

From Livro

Where Ano\_Aquisicao > 2000

**b)** Escreva o código SQL que permite responder à seguinte questão: nome e descrição do tipo de todas as obras impressas no ano de 1999.

Select Livro.Titulo, Tipo.Descricao

From Livro Inner Join Tipo On Livro.Cod\_Tipo = Tipo.Cod\_tipo

Where Livro.Ano\_impressao = 1999

**Nota**: no “Where” pode-se omitir o nome da tabela por este não existir em mais nenhuma. Também o Select livro-Titulo. Isto é, sempre que o campo seja único pode-se omitir o nome da tabela.