



DESENVOLVIMENTO DE SOFTWARE MULTIPLATAFORMA

Disciplina: IBD-016 – BANCO DE DADOS - NÃO RELACIONAL

Aula 1: Introdução ao Curso

Data 08/08/2024

Prof. Me. Anderson Silva Vanin

Quem sou eu

- Técnico em Eletrônica
- Bacharel em Ciência da Computação
- Pós Graduado em Banco de Dados
- Mestre em Gestão do Conhecimento e Informática (Aplicado a Visão Computacional)
- Atuação no CPS desde 2006
- Aulas nas disciplinas em diversas de Programação
- Cursos extracurriculares em Inteligência Artificial e IoT

Avaliações e Trabalhos

- **P1: 26/09/2024** (35%)

Avaliação Teórica/Prática em Laboratório de Informática

- **P2: 14/11/2024** (35%)

Avaliação Teórica/Prática em Laboratório de Informática

- **P3: 05/12/2024**

Avaliação Prática em Laboratório de Informática

- **T: 14/11/2024** - Trabalhos e atividades (30%)

Conjunto de Atividades solicitadas durante o semestre letivo somadas

CONTRATO PEDAGÓGICO

HORÁRIO DAS AULAS:

- **Início: 19h** (Se chegar atrasado, respeite os colegas e o professor: **ENTRE EM SILÊNCIO!**)
- Intervalo: 20h30
- Retorno do Intervalo: 21h00 (**RESPEITE O MESMO CRITÉRIO DE ENTRADA!**)
- **Saída: 22h30**

ATIVIDADES PRÁTICAS:

Entregas efetuadas pelo ambiente Teams.

Não serão aceitas entregas fora do prazo!

AVALIAÇÕES PRÁTICAS:

- As avaliações práticas podem ocorrer de forma remota (**Atente-se ao dia da avaliação e horário programado!**)

DURANTE AS AULAS NO LABORATÓRIO

- **NÃO CONSUMA NENHUM TIPO DE ALIMENTO NO LABORATÓRIO.**
- **NÃO SE DISTRAIA COM NENHUM OUTRO TIPO DE CONTEÚDO QUE NÃO AQUELE MINISTRADO EM AULA.**
- **NÃO INSTALE OU REMOVA QUALQUER OUTRO EQUIPAMENTO.**
- **MANTENHA O SILÊNCIO DURANTE A EXPLANAÇÃO DO PROFESSOR.**
- **NÃO CONVERSE ENQUANTO A AULA ESTIVER SENDO EXPLICADA!**
- **SE TIVER DÚVIDAS, PERGUNTE!**

Material e recursos para as aulas

- **GITHUB:**
https://github.com/ProfAndersonVanin/FATEC_BDNAORELACIONAL_2SEM2024
- **EMAIL:** anderson.vanin@fatec.sp.gov.br
- **LINKEDIN:** <https://br.linkedin.com/in/anderson-vanin>

Ementa da Disciplina

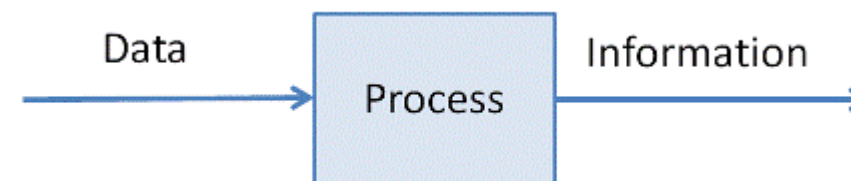
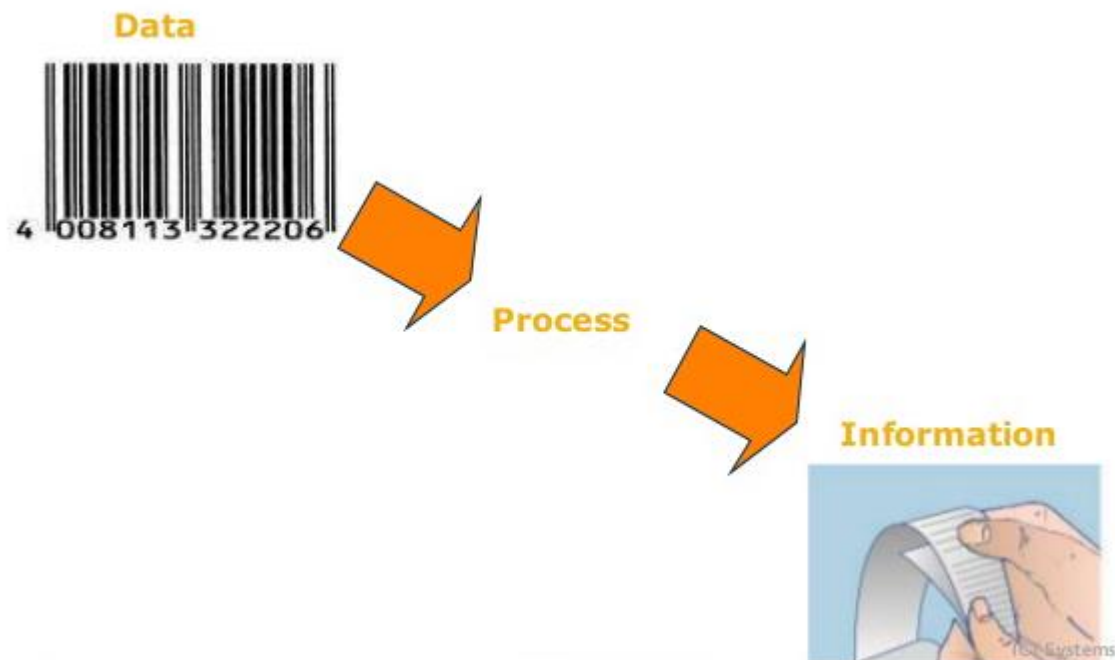
- Dados estruturados e não estruturados.
- Arquitetura de Bancos de Dados Não Convencionais.
- Introdução aos conceitos de Data Warehouse.
- Estudo sobre os conceitos de aplicações não-convencionais.
- Modelagem NoSQL: Definições e Motivação.
- Estudos das categorias de Bancos de Dados NoSQL: chave-valor, orientados a documentos, orientados a colunas e orientados a grafos. Projeto Lógico do Banco de Dados – Não Relacional.
- Implementações práticas das principais categorias de Bancos de Dados NoSQL.

Dados, Informação e Conhecimento

Dados, Informação e Conhecimento

Informação é o dado dentro de um contexto aplicável.

Uma informação requer um processamento ou uma interpretação (ou análise) de dados brutos.



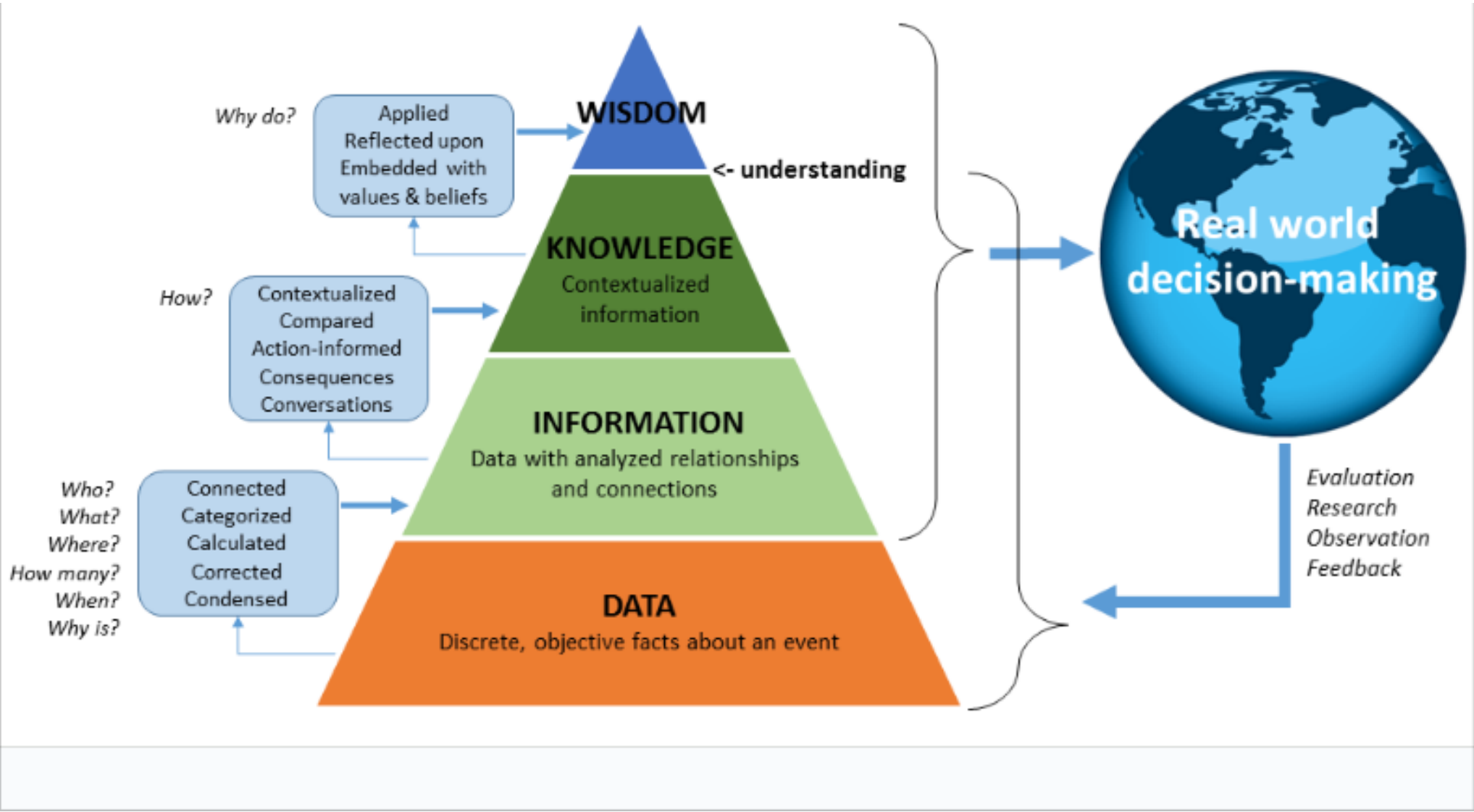
Dados, Informação e Conhecimento

Conhecimento é o processo de análise de uma informação e sua utilização para a tomada de decisão. Os dados correspondem ao nível mais baixo, depois de processados ou interpretados temos a informação e, então, quando ela é contextualizada, chegamos no conhecimento.

Dados, Informação e Conhecimento

Portanto, conhecimento é uma informação contextualizada, mas normalmente também corresponde a uma informação relevante e valiosa. Existe ainda mais um nível, quando o conhecimento é transformado em inteligência pelo processo do aprendizado ou aplicação e avaliação dos conhecimentos, servindo como importante ferramenta para tomada de decisões.

Dados, Informação e Conhecimento



Mercado hoje em dia

Áreas e cargos que trabalham diretamente (afinal, indiretamente, todas as áreas de TI atuam) com a manipulação dos dados nessa pirâmide evolutiva de inteligência estão em alta.

- O volume de dados gerados por usuários nunca foi tão grande.
- O volume de dados não estruturados vem crescendo muito nos últimos anos.
- O cruzamento de informações deixadas em redes sociais personaliza ofertas e visualizações.
- O avanço da Internet das Coisas (IoT) permite o cruzamento das informações com automações residenciais e industriais.

Dados Estruturados, Semi Estruturados e Não Estruturados



Dados Estruturados

Dados estruturados são aqueles organizados e representados com uma estrutura rígida, a qual foi previamente planejada para armazená-los.

Dados Estruturados

Pense em um formulário de cadastro com os campos: nome, e-mail, idade e uma pergunta que admite como resposta sim ou não. O campo nome será um texto, uma sequência de letras com ou sem a presença de espaços em branco, que terá um limite máximo e não poderá conter números ou símbolos. O campo e-mail também terá o padrão textual, mas formado por uma sequência de caracteres (e não só letras, pois admitirá números e alguns símbolos) e terá que ter obrigatoriamente um “@”. Idade é um campo que aceita apenas um número inteiro positivo, enquanto o campo referente a pergunta armazena um valor binário (pense um 1 bit, que pode ser 0 ou 1. Valor 0 para não, 1 para sim). Assim, cada campo possui um padrão bem definido, que representa uma estrutura rígida e um formato previamente projetado para ele.

Dados Estruturados

- Os dados de um mesmo cadastro estão relacionados (dizem respeito a mesma pessoa). Em outras palavras, os dados estruturados de um mesmo bloco (registro) possuem uma relação.
- Registros ou grupos de dados diferentes (como de pessoas diferentes), possuem diferentes valores, mas utilizam a mesma representação estrutural homogênea para armazenar os dados. Ou seja, possuem mesmo atributos (pense como sinônimo de campos no exemplo acima) e formatos, mas valores diferentes.

Dados Estruturados

O exemplo mais típico de dados estruturados é um banco de dados. Nele, os dados são estruturados conforme a definição de um esquema, que define as tabelas com seus respectivos campos (ou atributos) e tipos (formato).

O esquema pode ser pensado como uma metainformação do banco de dados, ou seja, uma descrição sobre a organização dos dados que serão armazenados no banco. É exatamente como no exemplo do formulário que, normalmente, está interligado com um banco de dados.

Dados Não Estruturados

Pense em um arquivo feito em um editor de texto. Você pode adicionar quanto texto quiser, sem se preocupar com campos, restrições e limites. O arquivo pode conter também imagens, como gráficos e fotos, misturado com textos. Imagens, assim como vídeos ou arquivos de áudio, são também exemplos de dados não estruturados.

Dados Não Estruturados

Assim, é fácil concluir que, por exemplo, as redes sociais, as quais possuem um enorme volume de dados, como textos, imagens e vídeos criados diariamente por usuários, representam outro exemplo de dados não estruturados. Atualmente, mais de 80% do conteúdo digital gerado no mundo é do tipo não estruturado.

Leiam este artigo:

<http://breakthroughanalysis.com/2008/08/01/unstructured-data-and-the-80-percent-rule/>

Dados Não Estruturados

Normalmente, basta pensar em uma situação de dados que não seguem estrutura para termos exemplos de dados não-estruturados, mas é preciso tomar um pouco de cuidado com essa análise.

Dados Não Estruturados

Em computação, todo dado, seja ele um arquivo ou um campo rígido, terá que ter algum tipo de estrutura, mesmo que mínima. Um arquivo é um tipo de estrutura mínima, pois é a unidade básica de armazenamento de um sistema operacional, mas ela é genérica, pois aceita diferentes tipos de dados.

Em resumo, quase tudo cairá em um arquivo, mesmo porque um vídeo tem que gravar em arquivo seus dados com um codificador (codec), um áudio também e assim por diante. Pensem, portanto, na estrutura interna do arquivo, se ela existe e é rígida, ou não.

Dados Estruturados, Semi Estruturados e Não Estruturados

Nome	País	Idade
Maria	Brasil	33
Miguel	México	45
John	Inglaterra	21
Louis	França	14

Dados Estruturados, Semi Estruturados e Não Estruturados

```
{  
  
  "pacotes-disponíveis": [  
  
    {  
  
      "disponível": "True",  
  
      "dias": 5,  
  
      "local": "Chile",  
  
      "preço-por-pessoa": 3000  
    },  
  
    {  
  
      "disponível": "False",  
  
      "dias": 6,  
  
      "local": "Argentina",  
  
      "preço-por-pessoa": 1000  
    },  
  
  ]  
}
```


Dados Estruturados, Semi Estruturados e Não Estruturados

Boa tarde,

Tudo bem? Acredito que houve um engano quanto à rota que o carro direcionando o Governador deveria seguir. O Governador João da Silva precisa visitar urgentemente as cidades do Sul de Minas Gerais, considerando que tem reuniões com os Prefeitos. Por isso, não faria sentido ele seguir em direção à Uberlândia, e sim em direção a Poços de Caldas depois do dia 10 de março de 2021.

Importante lembrar ele também visitará as cidades de Lavras (11 de março) e Varginha (13 de março).

Att.

Situação 1

“Ao final de um evento presencial, cadastrarei pessoas interessadas em fazer um curso online com uma empresa. Para reservar uma vaga, preciso do nome, e-mail e idade da pessoa.”

Solução A: Desenvolvi um sistema com campos em um formulário que cadastra os dados em um banco de dados. Planejei e criei previamente o banco de dados, o integrando com o sistema do formulário.

Solução B: Abri meu editor de texto e saí escrevendo os dados da galera. Coloquei cada dado em uma linha e tracei uma reta ao final de cada registro para separar as pessoas.

Situação 2

“Durante o cadastro, algumas pessoas acreditavam que o curso seria presencial, no lugar que tinha ocorrido o evento e não online, de forma que, para esse público, eles preferem aulas presenciais. Entrei em contato com meu cliente, a empresa que solicitou o cadastro dos interessados, e eles então me pediram que eu armazenasse o endereço de todos os interessados para saber se eles moram perto de alguma outra unidade da empresa.”

Situação 2

Solução A: Disse ao cliente que isso não seria possível. Não temos como inserir outra informação, como endereço, se não reformularmos o esquema do banco de dados para aceitar esse novo atributo, bem como editar a interface para inclusão de mais um campo, interligado ao atributo do banco. A estrutura de dados é rígida, por isso precisa ser previamente pensada e não será alterada na hora, em tempo real.

Solução B: Simples, criei uma linha a mais para cada registro e o editor de texto caiu como uma luva para essa demanda maluca em cima da hora. O arquivo de texto me permitiu qualquer tipo de cadastro, pois não impõe nenhuma estrutura interna rígida.

Situação 3

“O cliente deseja obter os dados em forma de relatório, com o número total de cadastros e cada registro em uma linha da planilha.””

Solução A: Simples, basta executar alguns comandos de banco de dados e teremos isso em segundos. O fato dos dados estarem estruturados ajuda muito tarefas como essa.

Solução B: Terei que contar manualmente, além de transpor manualmente para o formato de relatório também. Mesmo que eu faça um programa para fazer essa tarefa por mim, ainda assim, faremos uma busca em dados não estruturados.

Dados Semi-Estruturados

Apresentam uma representação heterogênea, ou seja, possuem estrutura, mas ela é flexível. Assim, pensando no exemplo acima, ela agrega um pouco dos dados lidos em termos de benefícios. Facilita o controle por ter um pouco de estrutura, mas também permite uma maior flexibilidade.

Dados Semi-Estruturados

Um exemplo típico é um arquivo em **XML** (eXtensible Markup Language, que significa, em português, linguagem de marcação estendida), o qual possui nós, que são rótulos de abertura e fechamento, este precedido com o símbolo “/”, com os dados inseridos entre os nós.

Dados Semi-Estruturados

Imagine o seguinte texto bruto (digo, por estar em um editor de texto):

Nome: Fulano da Silva Sauro

E-mail: fulano@internet.com.br

Rua Sei Lá, 1234, Mooca, São Paulo

Dados Semi-Estruturados

Agora, pensem nesses dados não-estruturados transpostos para um arquivo XML. O conteúdo do arquivo ficará assim:

```
<?xml version="1.0" encoding="UTF-8" ?>
```

```
<cadastro>
```

```
    <nome>Fulano da Silva Sauro</nome>
```

```
    <e-mail> fulano@internet.com.br </e-mail>
```

```
    <endereço>Rua Sei Lá, 1234, Mooca, São Paulo </endereço>
```

```
</cadastro>
```

Dados Semi-Estruturados

Outros exemplos de arquivos com dados semi-estruturados:

JSON – Javascript Object Notation (é um formato de arquivo padrão aberto e formato de intercâmbio de dados que usa texto legível para armazenar e transmitir objetos de dados que consistem em pares atributo-valor e arrays (ou outros valores serializáveis),

RDF – Resource Description Framework (modelo de dados para metadados - RDF é um grafo direcionado composto de declarações triplas),

OWL – Web Ontology Language (é uma linguagem da Web Semântica projetada para representar conhecimento rico e complexo sobre coisas).

XML - Extended Markup Language (é uma linguagem de marcação e formato de arquivo para armazenar, transmitir e reconstruir dados arbitrários. Ele define um conjunto de regras para codificar documentos em um formato legível por humanos e legível por máquina)

Resumo

Dados estruturados

Ex.: Banco de dados

Estrutura rígida
Projetada previamente
Representação homogêna

Cada campo de dados
tem um formato bem
definido.

Formato é um padrão
aceito pelo campo.

Dados de um mesmo
registro possuem
relação entre eles.

Registros possuem
valores diferentes, mas
mesmos atributos.

Atributos ou campos
são definidos por um
esquema.

Dados semi estruturados

Ex.: XML, JSON, RDF,
OWL.

Estrutura flexível
Representação heterogêna

Cada campo de dados
tem uma estrutura,
mas não existe uma
imposição de formato

O esquema é criado
com a definição de
elementos internos dos
arquivos (nós), legíveis
para seres humanos

Dados não estruturados

Ex.: Textos, arquivos,
documentos, imagens,
vídeos, áudios, redes
sociais etc.

Sem estrutura
(ou com estrutura mínima
de arquivo)

Mais de 80% dos
dados gerados no
mundo é deste tipo

Exercício de Revisão – Banco de Dados Relacional

Um berçário deseja informatizar suas operações. Quando um bebê nasce, algumas informações são armazenadas sobre ele, tais como: nome, data do nascimento, peso do nascimento, altura, a mãe deste bebê e o médico que fez seu parto. Para as mães, o berçário também deseja manter um controle, guardando informações como: nome, endereço, telefone e data de nascimento. Para os médicos, é importante saber: CRM, nome, telefone celular e especialidade.

Exercício de Revisão – Banco de Dados Relacional

1. Grife as entidades que você identificar no texto acima.
2. Liste as entidades que você grifou.
3. Liste os relacionamentos entre as entidades
 - **Exemplo: Mãe bebê: relacionamento entre Mãe e Bebê.**
4. Crie o modelo Entidade Relacionamento do exercício.
5. Crie o Modelo Físico do Exercício (SQL).

Respostas - Exercício de Revisão – Banco de Dados Relacional

Lista das entidades encontradas:

Um berçário deseja informatizar suas operações. Quando um bebê nasce, algumas informações são armazenadas sobre ele, tais como: *nome, data do nascimento, peso do nascimento, altura, a mãe* deste bebê e o *médico* que fez seu parto.

Para as mães, o berçário também deseja manter um controle, guardando informações como: *nome, endereço, telefone e data de nascimento*. Para os médicos, é importante saber: *CRM, nome, telefone celular e especialidade*.

Respostas - Exercício de Revisão – Banco de Dados Relacional

Liste os relacionamentos encontrados:

- **mãe_bebê**: relacionamento entre Mãe e Bebê.
- **Médico_bebê**: relacionamento entre Médico e Bebê

Respostas - Exercício de Revisão – Banco de Dados Relacional

Modelo Entidade Relacionamento

