

Instituto de Educação Superior da Paraíba

Curso: Sistemas de Informação Componente Curricular: Banco de Dados II

Semestre: 2019.1 Período: 5º Docente: Fábio Nicácio de Medeiros

Proposta de um Modelo de Banco de Dados Comercial na de Automação de Biblioteca

Eduardo Hollanda Leite

Igor Monteiro

Lucas Medeiros do Nascimento Pedrosa de Oliveira

João Pessoa-PB

2019

Eduardo Hollanda Leite

Igor Monteiro

Lucas Medeiros do Nascimento Pedrosa de Oliveira

Proposta de um Modelo de Banco de Dados Comercial na de Automação de Biblioteca

Projeto de Banco de Dados apresentado

ao Curso de Sistemas de informação, como um dos

critérios de avaliação do processo seletivo.

João Pessoa-PB

20191

Sumário

[Objetivos 4](#_Toc9788318)

[Metodologia do Trabalho 4](#_Toc9788319)

[1. Regras do Negócio 5](#_Toc9788320)

[1.1 Modelo Conceitual 8](#_Toc9788321)

[1.2 Modelo Lógico 10](#_Toc9788322)

[2. Dicionário 11](#_Toc9788323)

[3. RESTRIÇÕES (Constraints) 17](#_Toc9788324)

[4. Script 23](#_Toc9788325)

[5. Consultas 26](#_Toc9788326)

[6. Conclusão 27](#_Toc9788327)

[7. REFERÊNCIAS BIBLIOGRÁFICAS 28](#_Toc9788328)

Introdução

O sistema tem como finalidade, proporcionar o controle dos serviços realizados em uma Biblioteca, desde o cadastro de alunos, funcionários, cursos, fornecedores, editoras e livros; inserção de exemplares; solicitação de livros. Como também a realização de empréstimo, devolução e reserva de exemplares.

## Objetivos

Facilitar o controle de transações da Biblioteca, desde empréstimo, devolução e reserva de livros. Cadastro de alunos, funcionários, cursos, fornecedores, editoras e livros, inserção de exemplares; e solicitação de Livros. Realizará relatório de todos os exemplares que estão emprestados, exemplares atrasados informando o nome e matrícula do aluno, caso ele tenha a permissão as funcionalidades citadas.

## Metodologia do Trabalho

Este trabalho, foi realizado com a observação de funcionamento de uma Biblioteca, onde foi analisado todo o tramite dos serviços prestados, e pela deficiência que existe com a falta de controle dos processos administrativos e gerenciais da respectiva empresa.

Transformar os dados em informação é de suma importância para as organizações, onde pode preservar a vida da empresa. Nas primeiras observações realizadas, foi detectado a realização dos procedimentos realizados de forma manualmente e sem ter um sistema de automatização para facilitar, acelerar e assegurar os procedimentos.

# 1. Regras do Negócio

Regras de Negócio são declarações sobre a forma da empresa fazer [negócio](https://pt.wikipedia.org/wiki/Neg%C3%B3cio). Elas refletem políticas do negócio. As organizações com isto têm políticas para satisfazer os objetivos do negócio, satisfazer clientes, fazer bom uso dos recursos, e obedecer às leis ou convenções gerais do negócio. As regras do Negócio tornam-se requisitos, ou seja, podem ser implementados em um sistema de [software](https://pt.wikipedia.org/wiki/Software) como uma forma de requisitos de software desse sistema. Representam um importante conceito dentro do processo de definição de requisitos para [sistemas de informação](https://pt.wikipedia.org/wiki/Sistemas_de_informa%C3%A7%C3%A3o) e devem ser vistas como uma declaração genérica sobre a organização. As regras de negócio definem como o seu negócio funciona, podem abranger diversos assuntos como suas políticas, interesses, objetivos, compromissos éticos e sociais, obrigações contratuais, decisões estratégicas, leis e regulamentações entre outros.

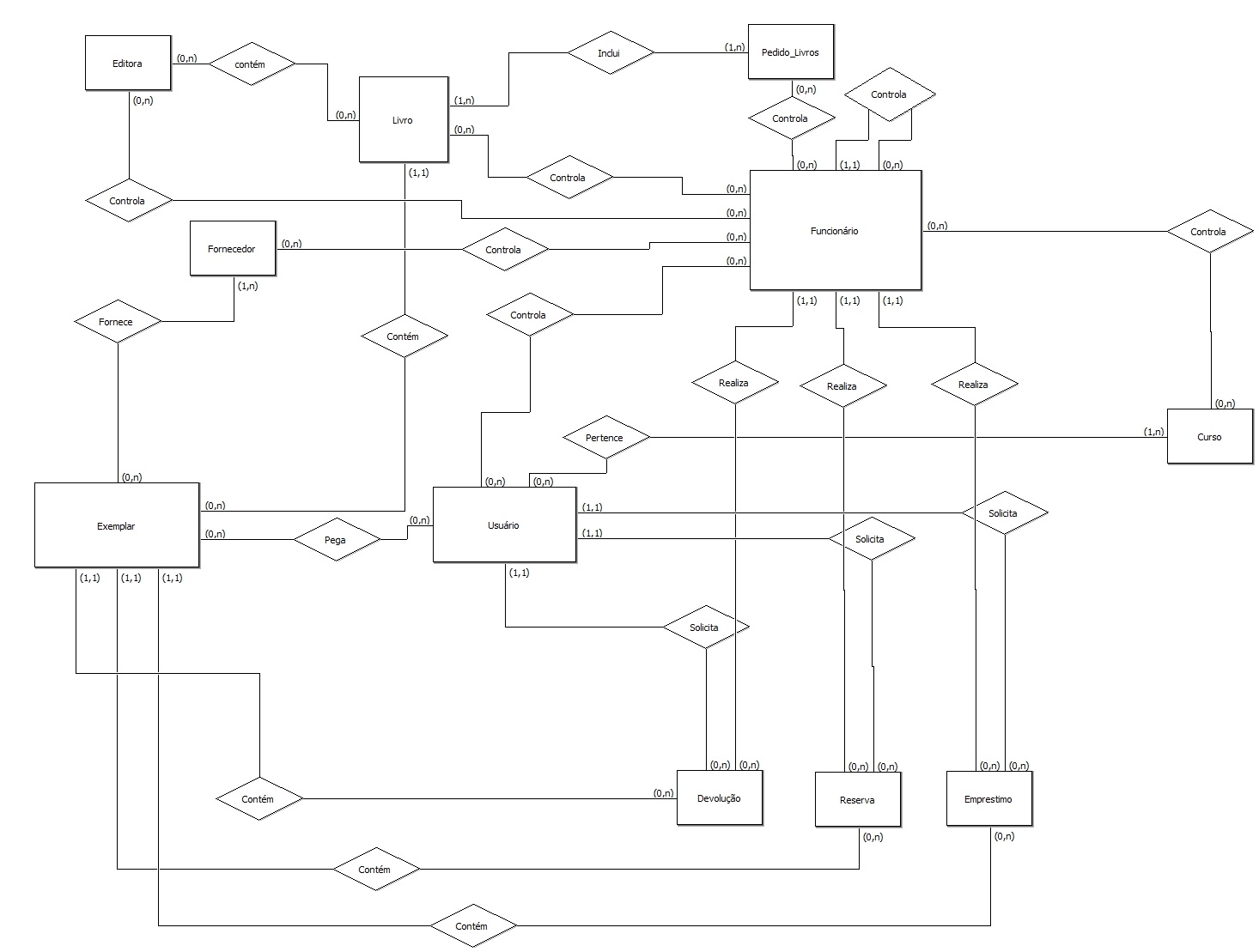
Diante a isso, a regra de negócio se aplica diretamente ao desenvolvimento de determinada plataforma de software voltada para um sistema de informação. É a regra de negócio que especifica as particularidades das funcionalidades a serem desenvolvidas. No processo de desenvolvimento de qualquer sistema, a regra de negócio visa detalhar as funcionalidades particulares do software. Com isso facilita por parte dos programadores o desenvolvimento de métodos de tratamento de exceções, particularidades que o sistema possa executar e o mais importante, limitar ações fora do processo normal de funcionamento de um sistema específico.

* Na Biblioteca, para cadastrar funcionários, será necessário informar (Nome, RG, CPF, E-mail, cargo, data de nascimento, telefone, salário, endereço, CEP, estado, cidade, bairro, senha, salário e o código do funcionário será gerado automaticamente).
* Qualquer funcionário cadastra aluno, sendo necessário o Usuário informar (Nome, RG, CPF, E-mail, data de nascimento, telefone, endereço, CEP, estado, bairro, cidade, senha, período, curso) e o funcionário deverá informar o tipo de usuário(professor, funcionário ou aluno) e a data de validade que é de 1 semestre(30/06) ou (31/12).
* Um funcionário pode cadastrar vários usuários, porém, um usuário pode ser cadastrado por um funcionário.
* Um aluno pode pegar emprestado até 5 livros, caso seja concluinte poderá pegar emprestado até 8 exemplares.
* O prazo de devolução de um livro será de 12 dias corridos, caso a data de devolução seja no final de semana ou feriado o sistema automaticamente deverá alterar a devolução para o próximo dia útil.
* Os funcionários terão diferentes tipos de permissões, limitadas de acordo com o seu cargo.
* Um funcionário poderá cadastrar vários fornecedores, porém, um fornecedor só poderá ser cadastrado por um funcionário.
* Para realizar o cadastro do fornecedor, o funcionário deverá informar (Nome do fornecedor, nome do responsável, CNPJ do fornecedor, E-mail, telefone, endereço, estado, bairro, CEP, cidade e o código do fornecedor deverão ser gerados automaticamente pelo sistema).
* Apenas alguns funcionários poderão incluir pedido de livros, na qual os funcionários que tiverem permissão poderão inserir vários pedidos, porém um pedido só poderá ser inserido por um funcionário.
* Para inserir um pedido, o funcionário deverá informar (Nome do fornecedor, o código do fornecedor, número da nota fiscal, data de emissão da nota fiscal, os títulos a serem inseridos com quantidade de exemplares e valor unitário e desconto de pedido caso tiver, o ano e a edição de cada livro, se não o sistema irá identificar automaticamente como 0% de desconto) o sistema deverá calcular a partir da quantidade de exemplares e valor unitário de cada título inseridos, o valor total do pedido e a quantidade total de títulos, também deverá gerar automaticamente o código do pedido).
* Caso um funcionário insira um livro que não tem cadastro no Banco, ele irá pegar as variáveis semelhantes entre Pedido de livro e livro, e inserir em livro, os demais campos serão cadastrados pelo funcionário assim que for finalizado o pedido.
* Apenas alguns funcionários poderão cadastrar livros, na qual os funcionários que tiverem permissão poderão cadastrar vários livros, porém o livro só poderá ser cadastrado por um funcionário. Após o cadastro, qualquer funcionário que tiver permissão poderá fazer alterações no cadastro.
* Para cadastrar um livro, o funcionário deverá inserir (CDU, Cutter, autores secundários em caso de ter no livro e nome da editora), o sistema deverá gerar automaticamente um registro bibliográfico e a data de cadastro do livro de acordo com a entrada do pedido e deverá pegar os demais campos a partir da tela de pedidos de livros.
* Apenas alguns funcionários poderão cadastrar Editora, na qual os funcionários que tiverem permissão poderão cadastrar várias editoras, porém uma editora só poderá ser cadastrada por um funcionário.
* Para cadastrar uma Editora, O funcionário deverá informar (Nome da editora, E-mail, telefone, celular, endereço, País, CEP, estado, cidade e bairro), o sistema deverá gerar automaticamente o código da editora.
* Apenas alguns funcionários poderão cadastrar Cursos, na qual os funcionários que tiverem permissão poderão cadastrar vários cursos, porém um curso só poderá ser cadastrado por um funcionário.
* Para cadastrar um curso o funcionário deverá informar (Nome do curso, quantidade de períodos e nome do coordenador), o sistema deverá gerar automaticamente o código de curso.
* Apenas alguns funcionários poderão inserir exemplares, na qual os funcionários que tiverem permissão poderão cadastrar vários exemplares, podem um exemplar só poderá ser cadastrado por um funcionário.
* Para cadastrar um exemplar o sistema deverá pegar os dados inseridos pelo funcionário na entidade cadastrar pedido de livros(Código do fornecedor, valor, número da nota fiscal, título e código de pedido de livros) e na entidade livro(Cutter, CDU e registro bibliográfico ), o sistema deverá gerar automaticamente o tombo de cada exemplar e o funcionário deverá inserir a situação do exemplar(disponível, empréstimo especial, extraviado ou desbastado) e observação caso seja necessário.
* Todos os funcionários terão permissão para realizar empréstimo, devolução e reserva.

## 1.1 Modelo Conceitual

**Modelo Conceitual** é um conjunto de suposições baseadas no mundo real que indicarão as **regas de negócio** de um sistema. Esta etapa independe da escolha de tecnologias e protótipos ajudam no entendimento dos processos. O modelo conceitual faz parte das primeiras atividades do projeto de interação. Com base na análise de requisitos e nas necessidades do usuário são definidas as estratégias para definição dos processos que estruturarão o produto. O**objetivo do modelo conceitual é criar um sistema coerente de objetos, propriedades e relações**claramente mapeados para o domínio da tarefa do usuário. Este cenário permite a utilização das chamadas **metáforas** que **estabelecem relação direta de uso ou reconhecimento entre um modelo**conhecido**do mundo real e uma parte do produto projetado**, seja na realização da tarefa ou na caracterização visual ou sonora do produto.

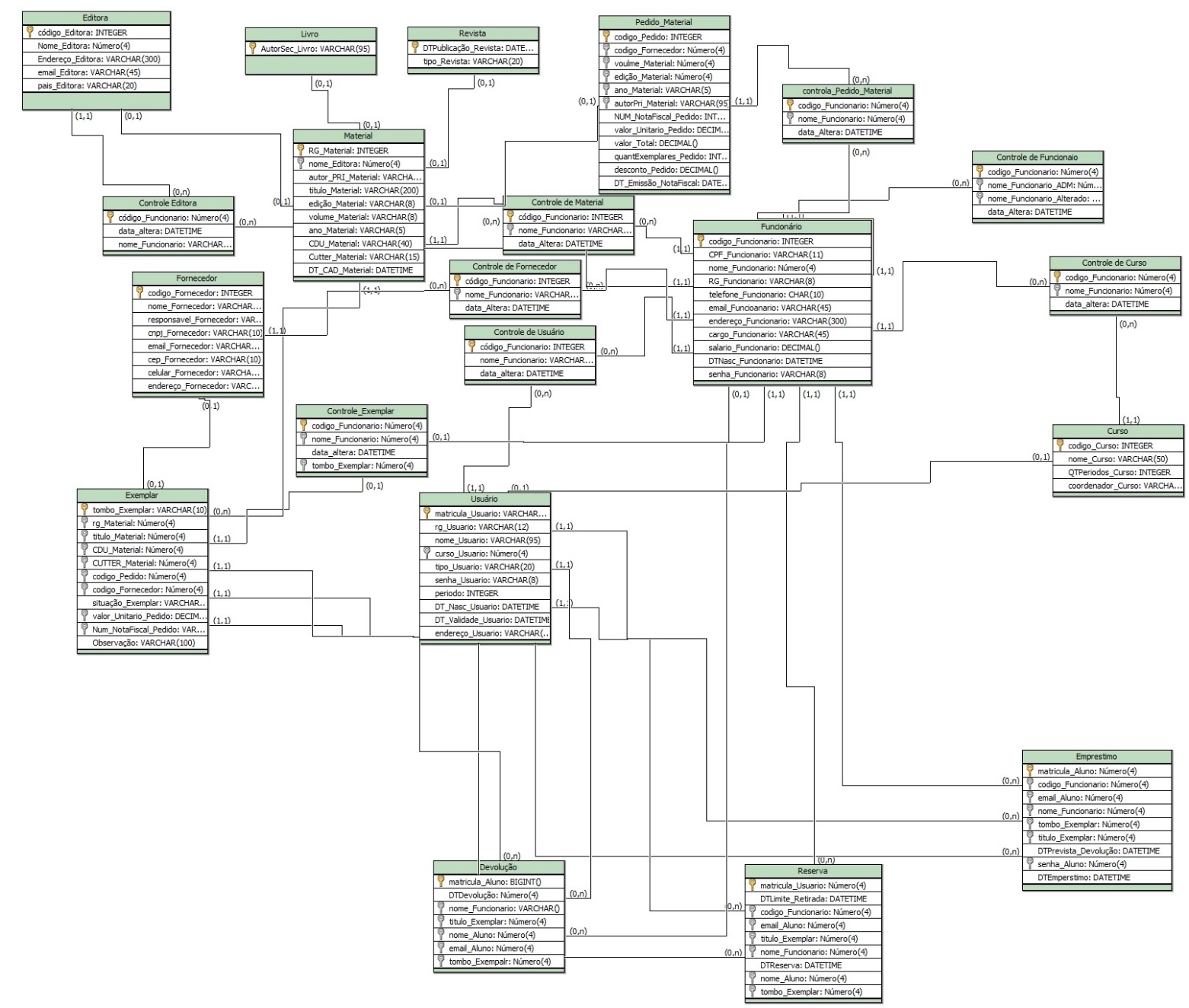
Todos os dados que podem aparecer em um banco de dados, são registrados pelo modelo conceitual. “Um modelo conceitual é uma descrição do banco de dados de forma independente de implementação em um SGBD." (Carlos A. Heuser, 1998, p 16). O medelo conceitual só não registra como os dados estão armazenados a nível de SGBD. Podemos compreender como a técnica mais conhecida de modelo conceitual a abordagem entidade-relacionamento (ER), onde o modelo conceitual é sempre representado através de um do diagrama chamado de diagrama entidade-relacionamento. Essa é a primeira, se não uma das fases mais importantes no projeto de construção de um banco de dados, “A saída dessa fase é uma especificação das necessidades do usuário" (Abraham Silberschatz, 2012, p 45).



## 1.2 Modelo Lógico

Um modelo de dados lógico é uma representação lógica das informações da área de negócios. Este é o conceito chave da modelagem de dados lógica. Compreende uma descrição das estruturas que serão armazenadas no banco e que resulta numa representação gráfica dos dados de uma maneira lógica, inclusive nomeando os componentes e ações que exercem uns sobre os outros. A técnica de modelagem mais difundida é a abordagem entidade-relacionamento (ER). Nesta técnica, um modelo conceitual é usualmente representado através de um diagrama, chamado diagrama entidade-relacionamento (DER).

Em nível de abstração, o modelo lógico é uma descrição de banco de dados no nível de abstração visto pelo usuário do SGBD. "O modelo lógico é dependente do tipo particular de SGBD que está sendo usado". (Carlos A. Heuser, 1998, p 17). Nesse modelo é descrito conforme vista de um usuário de SGBD a estrutura do banco de dados, detalhando o armazenamento interno de informações que não tem influência sobre a programação de aplicações, podendo assim influenciar a performance das aplicações. Dentro do modelo lógico é levando em conta algumas limitações e implementa recursos na forma adequada de padrão e nomenclatura, definindo assim as chaves primárias, estrangeiras, normalização, integridade referencial.



# 2. Dicionário

O dicionário de dados pode ser visto como um depósito central que descreve e define o significado de toda a informação usada na construção de um sistema, [metadados](https://pt.wikipedia.org/wiki/Metadados) que contém definições e representações de [elementos](https://pt.wikipedia.org/wiki/Elementos) de [dados](https://pt.wikipedia.org/wiki/Dados). Permite fazer a verificação de consistência entre os vários modelos. O seu conteúdo deve ser preciso, conciso e não redundante. Cada dicionário de dados deve ter no mínimo nome, tipo, descrição, pseudónimos, especificação e comentários significativos. Um dos benefícios de um dicionário de dados bem preparado é a consistência entre itens de dados através de diferentes tabelas.

Para explicar todos os objetos criados no modelo entidade-relacionamento é necessário um documento onde se tenha todas essas explicações, esse documento é o dicionário de dados. “O dicionário de dados consiste numa lista organizada de todos os elementos de dados que são pertinentes para o sistema”. (Carlos A. Heuser, 1998, p 21). Sem esse documento o modelo não pode ser considerado completo, pois descreve entradas, saídas, composição de depósitos de dados e alguns cálculos intermédios, onde consiste num ponto de referência de todos os elementos envolvidos na medida em que permite associar um significado a cada termo utilizado, permitindo que os analistas absorvam informações sobre todos os objetos do modelo da forma textual.

* **TblFuncionário**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ATRIBUTO** | **TIPO** | **NULO** | **DESCRIÇÃO** | **DOMÍNIO** | **CHAVE** | | |
| **PRI** | **EST** | **CAN** |
| Codigo\_Fucnionario | int | não | Código do Funcionario |  | **X** |  |  |
| Nome\_Funcionario | varchar(95) | não | Nome do Funcionário |  |  |  | X |
| Cpf\_Funcionario | varchar(11) | não | Cpf do Funcionario |  |  |  | X |
| Rg\_Funcioanro | varchar(8) | não | RG do Funcionario |  |  |  |  |
| Telefone\_Funcionario | varchar(10) | nãó | Telefone do Funcionario | xx xxxx-xxxx |  |  |  |
| Email\_Funcioanri | Varchar(45) | não | Email do Funcionario |  |  |  |  |
| Endereço\_Funcionario | Varchar(300) | não | Endereço do Funcionario |  |  |  |  |
| Cargo\_Funcionario | Varchar(45) | não | Cargo do Funcionario |  |  |  |  |
| Salario\_Funcionario | smallmoney | não | Salário do Funcionário |  |  |  |  |
| Senha\_Funcinario | Varchar(8) | não | Senha do Funcionario |  |  |  |  |
| Sexo\_Funcionario | Char(1) | não | Sexo do Funcionario | F ou M |  |  |  |

* **TblEditora**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ATRIBUTO** | **TIPO** | **NULO** | **DESCRIÇÃO** | **DOMÍNIO** | **CHAVE** | | |
| **PRI** | **EST** | **CAN** |
| Codigo\_Editora | int | não | Código da Editora |  | **X** |  |  |
| Nome\_Editora | varchar(50) | não | Nome da Editora |  |  |  | X |
| Email\_Editora | varchar(45) | não | Email da Editora |  |  |  |  |
| Pais\_Editora | varchar(20) | não | Pais da Editora |  |  |  |  |
| Endereço\_Editora | varchar(300) | nãó | Endereço da Editora |  |  |  |  |

* **TblFornecedor**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ATRIBUTO** | **TIPO** | **NULO** | **DESCRIÇÃO** | **DOMÍNIO** | **CHAVE** | | |
| **PRI** | **EST** | **CAN** |
| Codigo\_Fornecedor | int | não | Código do Fornecedor |  | **X** |  |  |
| Nome\_Fornecedor | varchar(50) | não | Nome do Fornecedor |  |  |  | X |
| Cnpf\_Fornecedor | varchar(10) | Sim | Cnpj do Fornecedor |  |  |  |  |
| Email\_Fornecedor | varchar(45) | sim | Email do Fornecedor |  |  |  |  |
| Cep\_Fornecedor | Varchar(10) | Sim | Cep do fornecedor | xxxxx-xxx |  |  |  |
| Cellular\_Fornecedor | Varchar(11) | Sim | Cellular do Fornecedor | xx xxxxx-xxxx |  |  |  |
| Endereço\_Forencedor | varchar(300) | não | Endereço do Fornecedor |  |  |  |  |

* **TblCurso**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ATRIBUTO** | **TIPO** | **NULO** | **DESCRIÇÃO** | **DOMÍNIO** | **CHAVE** | | |
| **PRI** | **EST** | **CAN** |
| Codigo\_Curso | int | não | Código do Curso |  | **X** |  |  |
| Nome\_Curso | varchar(50) | não | Nome do Curso |  |  |  | X |
| QTPeriodos\_Curso | int | não | Quantidade de Periodos do Curso |  |  |  |  |
| Coordenador\_Curso | varchar(95) | não | Nome do coordenador do cusro |  |  |  |  |

* **TblUsuario**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ATRIBUTO** | **TIPO** | **NULO** | **DESCRIÇÃO** | **DOMÍNIO** | **CHAVE** | | |
| **PRI** | **EST** | **CAN** |
| Matricula\_Usuario | Varchar(15) | não | Matrícula do Usuário |  | **X** |  |  |
| Rg\_Usuario | varchar(12) | não | Rg do Usuário |  |  |  | X |
| Nome\_Usuairo | Varchar(95) | não | Nome do Usuaário |  |  |  |  |
| Curso\_Usuario | varchar(95) | não | Codigo do Curso do Usuário |  |  | X |  |
| Tipo\_Usuario | Varchar(20) | Não | Tipo do Usuário |  |  |  |  |
| Email\_Usuario | Varchar(45) | Sim | Email do Usuário |  |  |  |  |
| Senha\_Usuario | Varchar(8) | Sim | Senha do Usuário |  |  |  |  |
| Sexo\_Usuario | Char(1) | Não | Sexo do Usuário | F ou M |  |  |  |
| periodo | int | não | Período que o Usuário está cursando |  |  |  |  |
| DT\_Nasc\_Usuario | datetime | não | Data de Nascimento do Usuário |  |  |  |  |
| DT\_Validade\_Usuário | datetime | não | Data limite da validade de acesso do Usuário |  |  |  |  |
| Endereço\_Usuairo | Varchar(300)m | sim | Endereço do Usuário |  |  |  |  |

* **TblMaterial**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ATRIBUTO** | **TIPO** | **NULO** | **DESCRIÇÃO** | **DOMÍNIO** | **CHAVE** | | |
| **PRI** | **EST** | **CAN** |
| RG\_Material | int | não | Registro Bibliográfico do Material |  | **X** |  |  |
| CodigoEditora | int | não | Código da Editora do Material |  |  | X |  |
| autorPri\_Material | Varchar(95) | não | Autor Principal do Material |  |  | X |  |
| Titulo\_Material | varchar(200) | Sim | Titutlo do Material |  |  |  |  |
| Edição\_Material | Varchar(8) | Sim | Edição do Material |  |  |  |  |
| Volume\_Material | Varchar(8) | sim | Volume do Material |  |  |  |  |
| ano\_Material | Varchar(5) | sim | Ano do Material |  |  |  |  |
| CDU\_Material | Varchar(40) | sim | CDU do Material |  |  |  |  |
| Cutter\_Material | Varchar(15) | sim | Cutter do Material |  |  |  |  |
| DT\_cad\_Material | datetime | não | Data de Cadastro do Material |  |  |  |  |

* **TblPedidoMaterial**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ATRIBUTO** | **TIPO** | **NULO** | **DESCRIÇÃO** | **DOMÍNIO** | **CHAVE** | | |
| **PRI** | **EST** | **CAN** |
| Código\_Pedido | int | não | Código do Pedido |  | **X** |  |  |
| Codigo\_Fornecedor | int | não | Código do Fornecedor do Pedido |  |  | X |  |
| codigo\_Material | Int | não | Código do Material a ser solicitado |  |  | X |  |
| Num\_NotaFiscal\_Pedido | Int | Não | Número da Nota Fiscal |  |  |  |  |
| Valor\_Unitario\_Pedido | smallmoney | não | Valor de cada Material |  |  |  |  |
| QuantExempares\_Pedido | int | não | Quantidade de Material |  |  |  |  |
| Desconto\_Pedido | smallmoney | sim | Desconto do Pedido |  |  |  |  |
| DT\_Emissão\_NotaFiscal | datetime | não | Data de Emissão da Nota Fiscal do Pedido |  |  |  |  |

* **TblExemplar**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ATRIBUTO** | **TIPO** | **NULO** | **DESCRIÇÃO** | **DOMÍNIO** | **CHAVE** | | |
| **PRI** | **EST** | **CAN** |
| Tombo\_Exemplar | Varchar(10) | não | Tombo do Exemplar |  | **X** |  |  |
| RG\_Material | int | não | Registro Bibliográfico do Exemplar |  |  | X |  |
| codigo\_Pedido | Int | não | Codigo do Peido do Exemplar |  |  | X |  |
| Codigo\_Fornecedor | Int | Não | Código do Fornecedor do Exemplar |  |  | X |  |
| Situação\_Exemplar | Char(1) | não | Situação que o exemplar se encontra |  |  |  |  |
| Num\_NotaFiscal\_Pedido | int | não | Número da Nota Fiscal |  |  |  |  |
| Obserção\_Exemplar | Varchar(100) | sim | Alguma observação do Exempalr |  |  |  |  |

* **TblEmprestimo**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ATRIBUTO** | **TIPO** | **NULO** | **DESCRIÇÃO** | **DOMÍNIO** | **CHAVE** | | |
| **PRI** | **EST** | **CAN** |
| Codigo\_Emprestimo | int | não | Código do Emprestimo |  | **X** |  |  |
| Matricula\_Usuario | Varchar(15) | não | Mátricula do Usuário |  |  | X |  |
| Codigo\_Funcionario | Int | não | Codigo do Funcionário |  |  | X |  |
| Tombo\_Exemplar | Varchar(10) | Não | Tombo do Exemplar |  |  | X |  |
| DTPrevista\_Devolução | datetime | não | Data Prevista de devolução |  |  |  |  |
| DTEmprestimo | datetime | não | Data de Emprestimo |  |  |  |  |

* **TblDevolução**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ATRIBUTO** | **TIPO** | **NULO** | **DESCRIÇÃO** | **DOMÍNIO** | **CHAVE** | | |
| **PRI** | **EST** | **CAN** |
| Codigo\_Devolução | int | não | Código da Devolução |  | **X** |  |  |
| Matricula\_Usuario | Varchar(15) | não | Mátricula do Usuário |  |  | X |  |
| Codigo\_Funcionario | Int | não | Codigo do Funcionário |  |  | X |  |
| Tombo\_Exemplar | Varchar(10) | Não | Tombo do Exemplar |  |  | X |  |
| DTDevolução | datetime | não | Data de Devolução |  |  |  |  |

* **TblReserva**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ATRIBUTO** | **TIPO** | **NULO** | **DESCRIÇÃO** | **DOMÍNIO** | **CHAVE** | | |
| **PRI** | **EST** | **CAN** |
| Codigo\_Reserva | int | não | Código da Reserva |  | **X** |  |  |
| Matricula\_Usuario | Varchar(15) | não | Mátricula do Usuário |  |  | X |  |
| Codigo\_Funcionario | Int | não | Codigo do Funcionário |  |  | X |  |
| Tombo\_Exemplar | Varchar(10) | Não | Tombo do Exemplar |  |  | X |  |
| DTReserva | datetime | não | Data de Reserva |  |  |  |  |
| DTLimite\_Reserva | Datetime | Não | Data Limite de Resrva |  |  |  |  |

* **TblRevista**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ATRIBUTO** | **TIPO** | **NULO** | **DESCRIÇÃO** | **DOMÍNIO** | **CHAVE** | | |
| **PRI** | **EST** | **CAN** |
| Titulo\_Revista | Varchar(200) | não | Título da Revista |  | **X** |  |  |
| Edição\_Revista | Varchar(8) | sim | Edição da Revista |  |  |  |  |
| Volume\_Revista | Varchar(8) | sim | Volume da Revista |  |  |  |  |
| DtPublicação\_Revista | Varchar(40) | Não | Data de Publicação da Revista |  |  |  |  |
| CDU\_Revista | Varchar(40) | não | CDU da Revista |  |  |  |  |
| Cutter\_Revista | Varchar(15) | Não | Cutter da Revista |  |  |  |  |

* **TblControla\_Editora**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ATRIBUTO** | **TIPO** | **NULO** | **DESCRIÇÃO** | **DOMÍNIO** | **CHAVE** | | |
| **PRI** | **EST** | **CAN** |
| Codigo\_Funcioanrio | int | não | Código do Funcionario que alterou |  |  |  |  |
| Codigo\_Editora | Int | não | Código da Editora |  |  |  |  |
| DT\_Altera | datetime | não | Data de Alteração |  | X |  |  |

* **TblControla\_Fornecedor**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ATRIBUTO** | **TIPO** | **NULO** | **DESCRIÇÃO** | **DOMÍNIO** | **CHAVE** | | |
| **PRI** | **EST** | **CAN** |
| Codigo\_Funcioanrio | int | não | Código do Funcionario que alterou |  |  |  |  |
| Codigo\_Fornecedor | Int | não | Código da Fornecedor |  |  |  |  |
| DT\_Altera | datetime | não | Data de Alteração |  | X |  |  |

* **TblControla\_Material**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ATRIBUTO** | **TIPO** | **NULO** | **DESCRIÇÃO** | **DOMÍNIO** | **CHAVE** | | |
| **PRI** | **EST** | **CAN** |
| Codigo\_Funcioanrio | int | não | Código do Funcionario que alterou |  |  |  |  |
| RG\_Material | Int | não | Código do Material |  |  |  |  |
| DT\_Altera | datetime | não | Data de Alteração |  | X |  |  |

* **TblControla\_Usuario**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ATRIBUTO** | **TIPO** | **NULO** | **DESCRIÇÃO** | **DOMÍNIO** | **CHAVE** | | |
| **PRI** | **EST** | **CAN** |
| Codigo\_Funcioanrio | int | não | Código do Funcionario que alterou |  |  |  |  |
| Matricula\_Usuario | Int | não | Matrícula do Usuário |  |  |  |  |
| DT\_Altera | datetime | não | Data de Alteração |  | X |  |  |

* **TblControla\_Pedido\_Material**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ATRIBUTO** | **TIPO** | **NULO** | **DESCRIÇÃO** | **DOMÍNIO** | **CHAVE** | | |
| **PRI** | **EST** | **CAN** |
| Codigo\_Funcioanrio | int | não | Código do Funcionario que alterou |  |  |  |  |
| Codigo\_Pedido\_Material | Int | não | Código do Pedido |  |  |  |  |
| DT\_Altera | datetime | não | Data de Alteração |  | X |  |  |

* **TblControla\_Pedido\_Material**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ATRIBUTO** | **TIPO** | **NULO** | **DESCRIÇÃO** | **DOMÍNIO** | **CHAVE** | | |
| **PRI** | **EST** | **CAN** |
| Codigo\_Funcioanrio\_alter | int | não | Código do Funcionario alterado |  |  |  |  |
| Codigo\_Funcioanrio\_Adm | Int | não | Código do Funcionario que alterou |  |  |  |  |
| DT\_Altera | datetime | não | Data de Alteração |  | X |  |  |

* **TblControla\_Curso**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ATRIBUTO** | **TIPO** | **NULO** | **DESCRIÇÃO** | **DOMÍNIO** | **CHAVE** | | |
| **PRI** | **EST** | **CAN** |
| Codigo\_Funcioanrio | int | não | Código do Funcionario que alterou |  |  |  |  |
| Codigo\_Curso | Int | não | Código do Curso |  |  |  |  |
| DT\_Altera | datetime | não | Data de Alteração |  | X |  |  |

# 3. RESTRIÇÕES (Constraints)

Constraints em tradução livre significa *“restrições",* é utilizado para manter os dados dos usuários restritos, evitando que dados inválidos sejam inseridos no banco de dados. Temos como exemplo a definição de um tipo de dado de uma coluna, essa definição nada mais é que uma constrains. Temos como as principais contraints: Not Null, Unique, Primary Key, Foreign Key e Default.

Not Null, impõe a uma coluna a NÃO aceitar valores NULL.

Unique, identifica de forma única cada registro em uma tabela de um banco de dados.

Primary Key ou chave Primária identifica de forma única cada registro em uma tabela de banco de dados.

Foreign Key, “(Chave Estrangeira) em uma tabela é um campo que aponta para uma chave primária em outra tabela. Desta forma, é usada para criar os relacionamentos entre as tabelas no banco de dados.” (Abraham Silberschatz, 2012, p 63).

Default, usada para inserir um valor padrão especificado em uma coluna.

* **TblFuncionario**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUNA** | **Tipo** | **NOME** | **EXPRESSÃO** |
| codigo\_Funcionario | Chave primária | PK\_TblFuncionario | PRIMARY KEY |
| Nome\_Funcionario | Chave Candidata | UN1\_TblFuncionario | UNIQUE |
| Cpf\_Funcionario | Chave Candidata | UN1\_TblFuncionario | UNIQUE |
| Sexo\_Funcionario | Validação do domínio., Caracteres permitidos:  F, M | CK\_TblFuncionario | check(sexo\_Funcionario like'[FM]') |

* **TblEditora**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUNA** | **Tipo** | **NOME** | **EXPRESSÃO** |
| codigo\_Editora | Chave primária | PK\_TblEditora | PRIMARY KEY |
| nome\_Editora | Chave Candidata | UN\_TblEditora | UNIQUE |

* **TblFornecedor**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUNA** | **Tipo** | **NOME** | **EXPRESSÃO** |
| codigo\_Fornecedor | Chave primária | PK\_TblFornecedor | PRIMARY KEY |
| nome\_Fornecedor | Chave Candidata | UN1\_TblFornecedor | UNIQUE |

* **TblCurso**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUNA** | **Tipo** | **NOME** | **EXPRESSÃO** |
| codigo\_Curso | Chave primária | PK\_TblCurso | PRIMARY KEY |
| nome\_Curso | Chave Candidata | UN\_TblCurso | UNIQUE |

* **TblUsuario**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUNA** | **Tipo** | **NOME** | **EXPRESSÃO** |
| matricula\_Usuario | Chave primária | PK\_TblUsuario | PRIMARY KEY |
| rg\_Usuario | Chave Candidata | UN1\_TblUsuario | UNIQUE |
| curso\_Usuario | Chave Estrangeira | FK\_TblUsuario\_TblCurso | FOREIGN KEY |
| sexo\_Usuario | Validação do domínio., Caracteres permitidos:  F, M | CK\_TblUsuario | check(sexo\_Usuario like'[FM]') |

* **TblMaterial**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUNA** | **Tipo** | **NOME** | **EXPRESSÃO** |
| RG\_Material | Chave primária | PK\_TblMaterial | PRIMARY KEY |
| codigo\_Editora | Chave Estrangeira | FK\_TblMaterial\_TblEditora | FOREIGN KEY |

* **TblPedidoMaterial**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUNA** | **Tipo** | **NOME** | **EXPRESSÃO** |
| codigo\_Pedido | Chave primária | PK\_TblPedidoMaterial | PRIMARY KEY |
| codigo\_Fornecedor | Chave Estrangeira | FK1\_TblPedidoMaterial\_TblFornecedor | FOREIGN KEY |
| codigo\_Material | Chave Estrangeira | FK2\_TblPedidoMaterial\_TblMaterial | FOREIGN KEY |
| Num\_NotaFiscal\_Pedido | Chave Candidata | UN1\_TblPedidoMaterial | UNIQUE |

* **TblExemplar**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUNA** | **Tipo** | **NOME** | **EXPRESSÃO** |
| tombo\_Exemplar | Chave primária | PK\_TblExemplar | PRIMARY KEY |
| RG\_Material | Chave Estrangeira | FK1\_TblExemplar\_TblMaterial | FOREIGN KEY |
| codigo\_Pedido | Chave Estrangeira | FK2\_TblExemplar\_TblPedidoMaterial | FOREIGN KEY |
| codigo\_Fornecedor | Chave Estrangeira | FK3\_TblExemplar\_TblFornecedor | FOREIGN KEY |
| Num\_NotaFiscal\_Pedido | Chave Estrangeira | FK4\_TblExemplar\_TblPedidoMaterial | FOREIGN KEY |
| situação\_Exemplar | Validação do domínio., Caracteres permitidos:  D, F, E, R, N | CK\_TblExemplar | check(situação\_Exemplar like'[DFERN]') |

* **TblEmprestimo**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUNA** | **Tipo** | **NOME** | **EXPRESSÃO** |
| codigo\_Emprestimo | Chave primária | PK\_TblEmprestimo | PRIMARY KEY |
| matricula\_Usuario | Chave Estrangeira | FK1\_TblEmprestimo\_TblUsuario | FOREIGN KEY |
| codigo\_Funcionario | Chave Estrangeira | FK2\_TblEmprestimo\_TblFuncionario | FOREIGN KEY |
| tombo\_Exemplar | Chave Estrangeira | FK3\_TblEmprestimo\_TblExemplar | FOREIGN KEY |

* **TblDevolução**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUNA** | **Tipo** | **NOME** | **EXPRESSÃO** |
| codigo\_Devolução | Chave primária | PK\_TblDevolução | PRIMARY KEY |
| matricula\_Usuario | Chave Estrangeira | FK1\_TblDevolução\_TblUsuario | FOREIGN KEY |
| codigo\_Funcionario | Chave Estrangeira | FK2\_TblDevolução\_TblFuncionario | FOREIGN KEY |
| tombo\_Exemplar | Chave Estrangeira | FK3\_TblDevolução\_TblExemplar | FOREIGN KEY |

* **TblReserva**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUNA** | **Tipo** | **NOME** | **EXPRESSÃO** |
| codigo\_Reserva | Chave primária | PK\_TblReserva | PRIMARY KEY |
| matricula\_Usuario | Chave Estrangeira | FK1\_TblReserva\_TblUsuario | FOREIGN KEY |
| codigo\_Funcionario | Chave Estrangeira | FK2\_TblReserva\_TblFuncionario | FOREIGN KEY |
| tombo\_Exemplar | Chave Estrangeira | FK3\_TblReserva\_TblExemplar | FOREIGN KEY |

* **TblRevista**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUNA** | **Tipo** | **NOME** | **EXPRESSÃO** |
| titulo\_Revista | Chave primária | PK\_TblRevista | PRIMARY KEY |

* **TblControla\_Editora**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUNA** | **Tipo** | **NOME** | **EXPRESSÃO** |
| DTAltera | Chave primária | PK\_TblControla\_Editora | PRIMARY KEY |
| codigo\_Funcionario | Chave Estrangeira | FK1\_TblControla\_Fornecedor\_TblFuncionario | FOREIGN KEY |
| codigo\_Editora | Chave Estrangeira | FK2\_TblControla\_Fornecedor\_TblFornecedor | FOREIGN KEY |

* **TblControla\_Fornecedor**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUNA** | **Tipo** | **NOME** | **EXPRESSÃO** |
| DTAltera | Chave primária | PK\_TblControla\_Fornecedor | PRIMARY KEY |
| codigo\_Funcionario | Chave Estrangeira | FK1\_TblControla\_Fornecedor\_TblFuncionario | FOREIGN KEY |
| codigo\_Fornecedor | Chave Estrangeira | FK2\_TblControla\_Fornecedor\_TblFornecedor | FOREIGN KEY |

* **TblControla\_Material**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUNA** | **Tipo** | **NOME** | **EXPRESSÃO** |
| DTAltera | Chave primária | PK\_TblControla\_Material | PRIMARY KEY |
| codigo\_Funcionario | Chave Estrangeira | FK1\_TblControla\_Fornecedor\_TblFuncionario | FOREIGN KEY |
| RG\_Material | Chave Estrangeira | FK2\_TblControla\_Fornecedor\_TblFornecedor | FOREIGN KEY |

* **TblControla\_Usuario**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUNA** | **Tipo** | **NOME** | **EXPRESSÃO** |
| DTAltera | Chave primária | PK\_TblControla\_Usuario | PRIMARY KEY |
| codigo\_Funcionario | Chave Estrangeira | FK1\_TblControla\_Usuario\_TblFuncionario | FOREIGN KEY |
| matricula\_Usuario | Chave Estrangeira | FK2\_TblControlaUsuario\_TblUsuario | FOREIGN KEY |

* **TblControla\_Pedido\_Material**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUNA** | **Tipo** | **NOME** | **EXPRESSÃO** |
| DTAltera | Chave primária | PK\_TblControla\_Pedido\_Material | PRIMARY KEY |
| codigo\_Funcionario | Chave Estrangeira | FK1\_TblControla\_Pedido\_Material\_TblFuncionario | FOREIGN KEY |
| codigo\_Pedido\_Material | Chave Estrangeira | FK2\_TblControla\_Pedido\_Material\_TblPedido\_Material | FOREIGN KEY |

* **TblControla\_Funcionario**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUNA** | **Tipo** | **NOME** | **EXPRESSÃO** |
| DTAltera | Chave primária | PK\_TblControla\_Funcionario | PRIMARY KEY |
| codigo\_Funcionario\_Alterado | Chave Estrangeira | FK1\_TblFucnionario\_Material\_TblFuncionario | FOREIGN KEY |
| codigo\_Funcionario\_Adm | Chave Estrangeira | FK2\_TblFucnionario\_Material\_TblFuncionario | FOREIGN KEY |

* **TblControla\_Curso**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUNA** | **Tipo** | **NOME** | **EXPRESSÃO** |
| DTAltera | Chave primária | PK\_TblControla\_Curso | PRIMARY KEY |
| codigo\_Funcionario | Chave Estrangeira | FK1\_TblControla\_Curso\_TblFuncionario | FOREIGN KEY |
| codigo\_curso | Chave Estrangeira | FK2\_ControlaCurso\_TblCurso | FOREIGN KEY |

# 4. Script

use master

GO

create database BD\_Biblioteca

ON(name=BDBiblio\_Date,

filename='C:\BDII\BDBiblioDate1.mdf',

size=50,

maxsize=1000,

filegrowth= 2)

LOG ON

(name= BDBiblio\_Log,

filename='C:\BDII\BDBiblioLog1.ldf',

size=500,

maxsize=5000,

filegrowth= 2)

GO

use BD\_Biblioteca

GO

create table TblFuncionario(

codigo\_Funcionario int constraint PK\_TblFuncionario primary key not null IDENTITY(1,1),

nome\_Funcionario varchar(95) constraint UN1\_TblFuncionario unique not null,

cpf\_Funcionario char(11) constraint UN2\_TblFuncionario unique not null,

rg\_Funcionario varchar(8) not null,

telefone\_Funcionario char(10),

email\_Funcionario varchar(45) not null,

endereço\_Funcionario varchar(300) not null,

cargo\_Funcionario varchar(45) not null,

salario\_Funcionario smallmoney not null,

DtNasc\_Funcionario datetime not null,

senha\_Funcionario varchar(8) not null,

sexo\_Funcionario char(1) not null constraint CK\_TblFuncionario check(sexo\_Funcionario like'[FM]'))

create table TblEditora(

codigo\_Editora int constraint PK\_TblEditora primary key not null IDENTITY(1,1),

nome\_Editora varchar(50) constraint UN\_TblEditora unique not null,

email\_Editora varchar(45) not null,

pais\_Editora varchar(20) not null,

endereco\_Editora varchar(300) not null)

create table TblFornecedor(

codigo\_Fornecedor int constraint PK\_TblFornecedor primary key not null IDENTITY(1,1),

nome\_Fornecedor varchar(50) constraint UN1\_TblFornecedor unique not null,

cnpj\_Fornecedor varchar(10),

email\_Fornecedor varchar(45),

cep\_Fornecedor varchar(10),

celular\_Fornecedor varchar(11),

endereço\_Fornecedor varchar(300) not null)

create table TblCurso(

codigo\_Curso int constraint PK\_TblCurso primary key not null IDENTITY(1,1),

nome\_Curso varchar(50) constraint UN\_TblCurso unique not null,

QTPeriodos\_Curso tinyint not null,

coordenador\_Curso varchar(95) not null)

create table TblUsuario(

matricula\_Usuario varchar(15) constraint PK\_TblUsuario primary key not null,

rg\_Usuario varchar(12) constraint UN1\_TblUsuario unique not null,

nome\_Usuario varchar(95) not null,

curso\_Usuario int constraint FK\_TblUsuario\_TblCurso foreign key references TblCurso(codigo\_Curso),

tipo\_Usuario varchar(20) not null,

email\_Usuario varchar(45),

senha\_Usuario varchar(8) not null,

sexo\_Usuario char(1) constraint CK\_TblUsuario check(sexo\_Usuario like'[FM]'),

periodo tinyint not null,

DT\_Nasc\_Usuario datetime not null,

DT\_Validade\_Usuario datetime not null,

endereço\_Usuario varchar(300))

create table TblMaterial(

RG\_Material int constraint PK\_TblMaterial primary key not null IDENTITY(1,1),

codigo\_Editora int constraint FK\_TblMaterial\_TblEditora foreign key references TblEditora (codigo\_Editora) not null,

autorPri\_Material varchar(95) not null,

titulo\_Material varchar(200),

edição\_Material varchar(8),

volume\_Material varchar(8),

ano\_Material varchar(5),

CDU\_Material varchar(40),

Cutter\_Material varchar(15),

DT\_Cad\_Material datetime not null)

create table TblPedidoMaterial(

codigo\_Pedido int constraint PK\_TblPedidoMaterial primary key IDENTITY(1,1) not null,

codigo\_Fornecedor int constraint FK1\_TblPedidoMaterial\_TblFornecedor foreign key references TblFornecedor(codigo\_Fornecedor) not null,

codigo\_Material int constraint FK2\_TblPedidoMaterial\_TblMaterial foreign key references TblMaterial(RG\_Material) not null,

Num\_NotaFiscal\_Pedido int constraint UN1\_TblPedidoMaterial unique not null ,

valorUnitario\_Pedido smallmoney not null,

quantExemplares\_Pedido int not null,

desconto\_Pedido smallmoney,

DT\_Emissão\_NotaFiscal\_Pedido datetime not null)

create table TblExemplar(

tombo\_Exemplar varchar(10) constraint PK\_TblExemplar primary key not null,

RG\_Material int constraint FK1\_TblExemplar\_TblMaterial foreign key references TblMaterial(RG\_Material) not null,

codigo\_Pedido int constraint FK2\_TblExemplar\_TblPedidoMaterial foreign key references TblPedidoMaterial(codigo\_Pedido) not null,

codigo\_Fornecedor int constraint FK3\_TblExemplar\_TblFornecedor foreign key references TblFornecedor(codigo\_Fornecedor) not null,

situação\_Exemplar char(1) not null constraint CK\_TblExemplar check(situação\_Exemplar like'[DFERN]'),--Disponível,Furtado,Emprestado,Restauração, Não Emprestar

Num\_NotaFiscal\_Pedido int constraint FK4\_TblExemplar\_TblPedidoMaterial foreign key references TblPedidoMaterial(Num\_NotaFiscal\_Pedido) not null, Observaçao\_Exemplar varchar(100))

create table TblEmprestimo(

codigo\_Emprestimo int constraint PK\_TblEmprestimo primary key IDENTITY(1,1) not null,

matricula\_Usuario varchar(15) constraint FK1\_TblEmprestimo\_TblUsuario references TblUsuario(matricula\_Usuario) not null,

codigo\_Funcionario int constraint FK2\_TblEmprestimo\_TblFuncionario foreign key references TblFuncionario(codigo\_Funcionario) not null,

tombo\_Exemplar varchar(10) constraint FK3\_TblEmprestimo\_TblExemplar foreign key references TblExemplar(tombo\_Exemplar) not null,

DTPrevista\_Devolução datetime not null,

DTEmprestimo datetime not null)

create table TblDevolução(

codigo\_Devolução int constraint PK\_TblDevolução primary key IDENTITY(1,1) not null,

matricula\_Usuario varchar(15) constraint FK1\_TblDevolução\_TblUsuario references TblUsuario(matricula\_Usuario) not null,

codigo\_Funcionario int constraint FK2\_TblDevolução\_TblFuncionario foreign key references TblFuncionario(codigo\_Funcionario) not null,

tombo\_Exemplar varchar(10) constraint FK3\_TblDevolução\_TblExemplar foreign key references TblExemplar(tombo\_Exemplar) not null,

DTDevolução datetime not null)

create table TblReserva(

codigo\_Reserva int constraint PK\_TblReserva primary key IDENTITY(1,1) not null,

matricula\_Usuario varchar(15) constraint FK1\_TblReserva\_TblUsuario references TblUsuario(matricula\_Usuario) not null,

codigo\_Funcionario int constraint FK2\_TblReserva\_TblFuncionario foreign key references TblFuncionario(codigo\_Funcionario) not null,

tombo\_Exemplar varchar(10) constraint FK3\_TblReserva\_TblExemplar foreign key references TblExemplar(tombo\_Exemplar) not null,

DTReserva datetime not null,

DTLimite\_Reserva datetime not null)

create table TblRevista(

titulo\_Revista varchar(200) constraint PK\_TblRevista primary key not null,

edição\_Revista varchar(8),

volume\_Revista varchar(8),

DTPublicaçao\_Revista varchar(30) not null,

CDU\_Revista varchar(40),

Cutter\_Revista varchar(15))

create table TblControla\_Editora(

codigo\_Funcionario int constraint FK1\_TblControla\_Editora\_TblFuncionario foreign key references TblFuncionario(codigo\_Funcionario) not null,

codigo\_Editora int constraint FK2\_TblControla\_Editora\_TblEditora foreign key references TblEditora(codigo\_Editora),

DTAltera datetime not null constraint PK\_TblControla\_Editora primary key)

create table TblControla\_Fornecedor(

codigo\_Funcionario int constraint FK1\_TblControla\_Fornecedor\_TblFuncionario foreign key references TblFuncionario(codigo\_Funcionario) not null,

codigo\_Fornecedor int constraint FK2\_TblControla\_Fornecedor\_TblFornecedor foreign key references TblFornecedor(codigo\_Fornecedor),

DTAltera datetime not null constraint PK\_TblControla\_Fornecedor primary key)

create table TblControla\_Material(

codigo\_Funcionario int constraint FK1\_TblControla\_Material\_TblFuncionario foreign key references TblFuncionario(codigo\_Funcionario) not null,

RG\_Material int constraint FK2\_TblControla\_Material\_TblMaterial foreign key references TblMaterial(RG\_Material),

DTAltera datetime not null constraint PK\_TblControla\_Material primary key)

create table TblControla\_Usuario(

codigo\_Funcionario int constraint FK1\_TblControla\_Usuario\_TblFuncionario foreign key references TblFuncionario(codigo\_Funcionario) not null,

matricula\_Usuario varchar(15) constraint FK2\_TblControlaUsuario\_TblUsuario foreign key references TblUsuario(matricula\_Usuario),

DTAltera datetime not null constraint PK\_TblControla\_Usuario primary key)

create table TblControla\_Pedido\_Material(

codigo\_Funcionario int constraint FK1\_TblControla\_Pedido\_Material\_TblFuncionario foreign key references TblFuncionario(codigo\_Funcionario) not null,

codigo\_Pedido\_Material int constraint FK2\_TblControla\_Pedido\_Material\_TblPedido\_Material foreign key references TblPedidoMaterial(codigo\_Pedido),

DTAltera datetime not null constraint PK\_TblControla\_Pedido\_Material primary key)

create table TblControla\_Funcionario(

codigo\_Funcionario\_Alterado int constraint FK1\_TblFucnionario\_Material\_TblFuncionario foreign key references TblFuncionario(codigo\_Funcionario) not null,

codigo\_Funcionario\_Adm int constraint FK2\_TblFucnionario\_Material\_TblFuncionario foreign key references TblFuncionario(codigo\_Funcionario) not null,

DTAltera datetime not null constraint PK\_TblControla\_Funcionario primary key)

create table TblControla\_Curso(

codigo\_Funcionario int constraint FK1\_TblControla\_Curso\_TblFuncionario foreign key references TblFuncionario(codigo\_Funcionario) not null,

codigo\_curso int constraint FK2\_ControlaCurso\_TblCurso foreign key references TblCurso(codigo\_Curso),

DTAltera datetime not null constraint PK\_TblControla\_Curso primary key)

# 5. Consultas

* **TblFuncionario**

Mostrar todos atributos da tabela funcionario

SELECT \* FROM funcionário

Mostrar todos os nomes dos funcionario

SELECT nome FROM funcionário

Mostrar todos os funcionários que recebem mais de R$ 1.500

SELECT \* FROM funcionario WHERE salario < 1500

Mostrar funcionário Homens

SELECT \* FROM funcionario WHERE SEXO = ‘M’

Mostrar todos os CPFs dos fucionario

SELECT cpf FROM funcionário

Mostrar todos os RGS dos fucionario

SELECT Rg FROM funcionario

Mostrar todos no cargo de supervisor

SELECT \* FROM funcionario WHERE cargo = ‘supervisor’

Mostrar todos os homens que começam com a letra A

SELECT \* FROM funcionario WHERE (sexo = ‘m’) and (nome like ‘a%’)

Mostrar todos os funcionário so sexo feminino

SELECT \* FROM funcionario WHERE SEXO = ‘F’

Mostrar quantos funcionarios

SELECT COUNT(codigo) FROM funcionario

* **TblEditora**

Mostrar todos atributos da tabela funcionario

SELECT \* FROM editora

Mostrar todas as editoras que estão no brasil

SELECT \* FROM editora WHERE pais = ’brasil’

Mostrar todas as editoras que tem como endereço Manaira

SELECT \* FROM editora WHERE endereço = ‘manaira’

Mostra quantas editoras estão cadastradas

SELECT COUNT(codigo) FROM editora

Mostra todas editoras que começam com A

SELECT \* FROM editora WHERE nome = ‘a%’

Mostra as editoras que estão em países com a letra B

SELECT \* FROM editora WHERE pais = ‘B%’

Mostra todas as editoras que não preencheram o campo de email

SELECT \* FROM editora WHERE email IS NULL

Mostras todas as editoras que estão cadastradas como gmail

SELECT \* FROM editora WHERE email = ‘%gmail.com’

Mostrar todas os países onde tem editoras cadastradas

SELECT pais FROM editora

Mostra todos os endereços onde tem editoras cadastradas

SELECT endereco FROM editora

* **TblFornecedor**

Mostrar todos atributos da tabela fornecedor

SELECT \* FROM fornecedor

Mostrar quantos fornecedores estão cadastrados

SELECT COUNT(codigo) FROM fornecedor

Mostrar todos os fornecedores

SELECT nome FROM fornecedores

Mostrar todos os fornecedores que estão com email cadastrados com gmail

SELECT email FROM fornecedores WHERE email = ‘%gmail.com’

Mostrar todos os endereços onde tem fornecedores

SELECT endereco FROM fornecedor

Mostrar todos os CNPJs dos fornecedores

SELECT cnpj FROM fornecedor

Mostrar todos os celulares dos fornecedores

SELECT celular FROM fornecedor

Mostrar todos fornecedores que estão sem cep cadastrados

SELECT \* from fornecedor where cep is NULL

Mostrar todos os fornecedores que tem o cep começados em 58

SELECT \* from fornecedor where cep = ‘58%’

* **TblCurso**

Mostrar todos atributos da tabela curso

SELECT \* FROM curso

Mostrar a quantidade de cursos existentes

SELECT COUNT(codigo) FROM curso

Mostrar todos os cursos

SELECT nome FROM cursos

Mostrar os coordenadores

SELECT coordenador FROM curso

Mostrar os coordenadores que comecem com a letra C

SELECT \* FROM curso WHERE coordenador = ‘C’

Mostrar quantos períodos tem o curso de SI

SELECT OTPeriodos FROM curso where cuso = ‘SI’

Mostrar todos os cursos que começem com a letra M

SELECT \* FROM cursos WHERE nome = ‘m%’

Mostrar todos os cursos que tem menos de 7 periodos

SELECT \* FROM curso WHERE OTPeriodos < 7

Ordenar todos os cursos do mais novos aos mais antigos

SELECT nome FROM curso ORDER BY codigo

Ordenar os cursos em ordem alfabetica

SELECT SELECT nome FROM curso ORDER BY nome

* **TblUsuario**

Mostrar todos atributos da tabela Usuario

SELECT \* FROM usuairo

Mostrar os usuario por ordem alfabética

SELECT nome FROM usuário ORDER BY nome

Mostrar os alunos que estão no sétimo periodo

SELECT \* from usuário WHERE período = 7

Mostrar todos os usuriarios do sexo masculino

SELECT \* from usuário WHERE sexo = ‘M’

Mostrar todos os que moram em Joao Pessoa

SELECT \* FROM usuário WHERE endereço = ‘joao pessoa’

Mostrar todos os tipos de usuairos

SELECT tipos FROM usuario

Mostrar todos os usuário que estão sem preencher os RG no cadastro

SELECT \* FROM usuário WHERE rg IS NULL

Mostrar todos os usuariario que cadastraram o email com hotmail

SELECT \* from usuário WHERE email = ‘% Hotmail.com’

Mostrar os usuários por ordem de idade

SELECT \* from usuário ORDER BY UNIX\_TIMESTAMP (DT\_Nasc)

* **TblMaterial**

Mostrar todos os atributos da tabela material

SELECT \* FROM material

Mostrar todos os materiais , do mais novo pra o mais antigo

SELECT \* FROM material ORDER BY ano ASC

Mostrar todos os materiais que tenha como autor Machado de Assis

SELECT \* FROM material WHERE autorPri = ‘Machado de Assis’

Mostrar todos títulos dos matérias por ordem alfabética

SELECT titulo FROM material ORDER BY titulo

Mostrar os matérias mais recentes cadastrados

SELECT \* FROM material ORDER BY UNIX\_TIMESTAMP (DT\_cad)

Mostrar todos os materiais que tenham mais de 2 volumes

SELECT \* FROM material WHERE volume > 2

Mostrar todos os materiais ordenados por ordem alfabética de edição

SELECT \* FROM material ORDER BY edicao

Mostrar todas as editoras ordenadas por codigo

SELECT \* FROM material ORDER BY codigoEDitora

Mostrar os matérias cadastrados do mais novo para o mais antigo

SELECT \* FROM material ORDER BY ano

Mostrar todos os matérias do ano atual

SELECT titulo FROM material WHERE ano = ‘2019’

* **TblPedidoMaterial**

Mostrar todos os atributos da tabela pedidomaterial

SELECT \* FROM pedidoMaterial

Mostrar todos os pedidos ordenados por codigo

SELECT \* FROM pedidoMaterial ORDER BY código\_pedido

Mostrar todos os pedidos ordenados pela quantidade

SELECT \* FROM pedidoMaterial ORDER BY QuantExempares

Mostrar todos pedidos que tenham valor a cima de 100 reais

SELECT \* from pedidoMaterial WHERE Valor\_Unitario > 100

Listar todos os pedidos em ordem por preço

SELECT \* FROM pedidoMaterial ORDER BY Valor\_Unitario

Mostrar todos os pedidos em ordem por Num\_NotaFiscal

SELECT \* FROM pedidoMaterial ORDER BY Num\_NotaFiscal

Mostrar todos os pedidos que tenham data de emissão desse ano

SELECT \* FROM pedidoMaterial WHERE DT\_Emissão\_NotaFisca > 2019

Mostrar a quantidade de fornecedores

SELECT COUNT(código\_fornecedor) FROM curso

Mostrar todos os pedidos que tem desconto

SELECT \* FROM pedidoMaterial WHERE desconto\_pedido IS NOT NULL

* **TblExemplar**

Mostrar todos os atributos da tabela exemplar

SELECT \* FROM exemplar

Mostrar mostrar quantos exemplares existem

SELECT COUNT(codigo) FROM exemplar

Mostrar todos os exemplares que tem tombo cadastrado

SELECT \* FROM exemplar WHERE tombo IS NOT NULL

Mostrar todas as observações de cada exemplar

SELECT observacao\_exemplar FROM exemplar

Mostrar todos os exemplares por ordem de registro

SELECT \* FROM exemplar ORDER BY rg\_material

Mostrar todos os exemplares por ordem de pedido

SELECT \* FROM exemplar ORDER BY código\_pedido

Mostrar a situação de todos os exemplares

SELECT situacao\_exemplar FROM exemplar

Mostrar o numero de nota fiscal de todos exemplares

SELECT Num\_NotaFiscal FROM exemplar

Mostrar código de pedido e do fornecedor

SELECT código\_pedido , código\_Fornecdor FROM exemplar

Mostrar quantas notas ficais foram feitas

SELECT COUNT(Num\_NotaFiscal) FROM exemplar

* **TblEmprestimo**

Mostrar todos os atributos da tabela emprestimo

SELECT \* FROM emprestimo

Mostrar quantos usuário estão matriculados

SELECT COUNT(matricula\_Usuario) FROM emprestimo

Mostrar quantos livros foram emprestados

SELECT COUNT(código\_emprestimo)FROM empréstimo

Mostrar todas as datas de devolução

SELECT DTEmprestio FROM emprestimo

Mostrar quantos livros estão com tombamento

SELECT COUNT(tombo\_exemplar) FROM emprestimo

Mostrar todos os tombamentos maiores que 500

SELECT tombo\_exemplar WHERE tombo\_exemplar > 500

Mostrar quantos funcionários estão cadastrados

SELECT COUNT(funcionario) FROM emprestimo

Mostrar todos os empréstimos por ordem crescente

SELECT codigo\_emprestimo FROM empréstimo ORDER BY codigo\_emprestimo

Mostrar código de funcionário por ordem crescente

SELECT codigo\_funcioanario FROM empréstimo ORDER BY codigo\_funcionario

Mostrar todas as datas previstas para devolução

SELECT DTPrecista\_Devolucao FROM empréstimo

* **TblDevolução**

Mostrar todos atributos da tabela devolucao

SELECT \* FROM devolucao

Mostrar todos os códigos de devolucao

SELECT código\_devolucao FROM devolucao

Mostrar usuário com matricula 500

SELECT matricula\_usuario FROM devolução WHERE matricula\_Usuario = 500

Mostrar todos os funcionários por ordem crescente do codigo

SELECT código\_funcionário FROM devolução ORDER BY código\_funcionario

Mostrar todas as datas de devolução

SELECT DTDevolucao FROM devolucao

Mostrar quantos livros estão com tombamento

SELECT COUNT(tombo\_exemplar) FROM devolucao

Mostrar todos os livos que tem devolução entre o dia 16 do 4 e o dia 27 do 8

SELECT \* FROM devolução WHERE Status = 1 AND DATE (Data) > ‘2018-04-016’ AND DATE (Data) < ‘2018-08-27’

Mostrar todos os funcionários que tenham código entra 120 e 170

SELECT código\_funcionario WHERE código\_Funcionario BETWEEN 120 AND 170

Mostrar todos os livros por ondem crescente de seu tombo

SELECT tombo\_exemplar FROM devolução ORDER BY tombo\_exemplar

* **TblReserva**

Mostrar todos os atributos da tabela reserva

SELECT \* from Reserva

Mostrar todas as matriculas dos usuários cadastrados

SELECT matricula\_usuairo FROM reserva

Mostrar os usuairo que foram cadastrados entre o dia 1 ao dia 15

SELECT \* FROM devolução WHERE Status = 1 AND DATE (Data) > ‘2018-04-01’ AND DATE (Data) < ‘2018-04-15’

Mostrar todas as datas limites de reserva

SELECT DTLimite FROM reserva

Mostrar os códigos de reserva por ordem crescente

SELECT codigo FROM reserva ORDER BY código ASC

Mostrar os códigos de reserva por ordem decrescente

SELECT codigo FROM reserva ORDER BY código DESC

Mostrar quantos livros foram reservados no dia 25 do 4

SELECT COUNT(DTReserva = ‘2019-04-25’) FROM emprestimo

Mostrar matricula do usuario de forma decrescente

SELECT matricula\_usuario FROM reserva ORDER BY matricula\_usuario

* **TblRevista**

Mostrar todos atributos da tabela revista

SELECT \* FROM Revista

Mostrar todos os edicao de revista

SELECT edicao FROM revista

Mostrar todos os cutter de forma crescente

SELECT cutter FROM revista ORDER BY cutter

Mostrar todos os títulos das revista que tem volume maior que 3

SELECT titulo FROM revista WHERE volume > 3

Mostrar a edicao das revistas em ordem decrescente

SELECT edicao FROM revista ORDER BY edicao

Mostrar data da publicação maior que 2000

SELECT DtPublicacao FROM revista WHERE DtPublicacao < 2000

Mostrar CDU de todas as revistas

SELECT cdu FROM revistas

Mostrar edicao das revistas , das mais novas para as mais velhas

SELECT edicao FROM revista ORDER BY DtPublicacao

Mostrar todos os títulos que começam com a letra a

SELECT titulo FROM revista WHERE titulo = ‘a%’

Mostrar todos os títulos que começam com a letra B e tem volume maior que 2

SELECT SELECT titulo FROM revista WHERE titulo = ‘b%’ and volume > 2

* **TblControla\_Material**

Mostrar todos os atributos da tabela controla\_material

SELECT \* FROM controla\_material

Mostrar mostrar códigos de funcionário em ordem decrescente

SELECT código\_funcionário FROM controla\_material ORDER BY código\_funcionario

Mostrar as datas alteradas de 2005 ate hoje

SELECT \* FROM devolução WHERE Status = 1 AND DATE (Data) > ‘2005-01-01’

Mostrar o código dos matérias entre 5 e 250

SELECT código\_material FROM controla\_material WHERE código\_materia BETWEEN 5 AND 250

Mostrar os rg dos matérias que foram alterados entre o ano de 2018 a 2019

SELECT rg\_material FROM devolução WHERE Status = 1 AND DATE (Data) > ‘2018-01-01’ AND DATE (Data) < ‘2019-01-01’

Mostrar todas as datas de alteração por ordem crescente

SELECT DT\_altera from controla\_usuario ORDER BY DT\_Altera ASC

Mostrar todas as datas de alteração por ordem decrescente

SELECT DT\_altera from controla\_usuario ORDER BY DT\_Altera DESC

# 6. Conclusão

No final desse trabalho podemos compreender a importância de um banco de dados em uma empresa, o banco veio para facilitar e maximizar os ganhos da empresa, transformando os dados colocados neles em informações fazendo com que a empresa consiga fazer um planejamento melhor e focar em objetivos mais expressivos, podendo tirar muitas vantagens em relações as outras. Podemos tomar como exemplo o meio que trabalhamos , no ramo na biblioteca é preciso se ter muito controle sobre todos os dados presentes tendo em vista o numero de dados que devem ser armazenados(livros e cada usuário), fazendo o uso do banco nos economizamos varias horas de trabalho ,tendo em vista que seria necessário armazenar os dados manualmente em papei que ainda corriam o risco de se perder, economizando tempo e dinheiro.

# 7. REFERÊNCIAS BIBLIOGRÁFICAS

HEUSER ALBERTO, Carlos. Projeto de Banco de Dados: 6. ed. São Paulo: Bookman Editora, 1998.

SILBERSCHATZ, Abraham. Sistema de Banco de Dados. 6 ed. Rio de Janeiro: Elsevier Editora*,* 2002.