

Exercício 1

Analise o histórico de um dos alunos de uma faculdade.

Universidade Federal do Rio Grande do Norte				
Curso de: Ciências da Computação		Código do Curso: 0037		
Aluno: Victor Alexandre da Silva		Matricula: 007043	Status: Regular	
Histórico				
Disciplina (código)	Professor - Código	Nota	Faltas	Situação
Análise de sistemas (AN001)	Roberto Carlos - 001	7.5	7	Aprovado
Matemática (MA002)	Jandira - 002	8.5	6	Aprovado
Inglês (IN101)	Junior Villas - 003	4.5	0	Reprovado

A partir do histórico apresentado acima, utilize a ferramenta *MySQL Workbench* para:

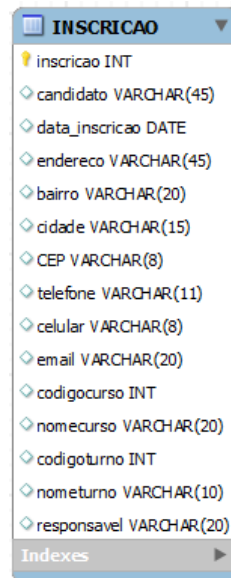
- Apresentar uma tabela que represente a entidade não normalizada (0FN) *Aluno*.
- Aplique a primeira forma normal (1FN) na tabela criada, justificando suas modificações. Você deverá manter a tabela original (na 0FN) para que seja possível realizar a comparação.
- Aplique a segunda forma normal (2FN) na tabela criada, justificando suas modificações. Mantenha as tabelas 0FN e 1FN para que possa comparar com as tabelas na 2FN.

Exercício 2

Observe a ficha de inscrição abaixo. Refere-se à inscrição de processo seletivo de uma instituição de um dos candidatos.

Federação de Escolas Faculdades Integradas Simonsen					
Ficha de Inscrição Processo Seletivo					
Inscrição : 00010	Candidato : Robson Vieira		Data Inscrição: 03/06/2008		
Endereço :	Bairro :	Cidade :	Cep:		
Telefone :	Celular :	email:			
Código Curso:	Nome do Curso:	Código Turno:	Nome Turno:		
Responsável (caso seja menor de idade):					

A tabela abaixo representa a entidade *Inscrição*.



INSCRICAO	
inscricao	INT
candidato	VARCHAR(45)
data_inscricao	DATE
endereco	VARCHAR(45)
bairro	VARCHAR(20)
cidade	VARCHAR(15)
CEP	VARCHAR(8)
telefone	VARCHAR(11)
celular	VARCHAR(8)
email	VARCHAR(20)
codigocurso	INT
nomecurso	VARCHAR(20)
codigoturno	INT
nometurno	VARCHAR(10)
responsavel	VARCHAR(20)

Utilize a ferramenta *MySQL Workbench* e com base nos conhecimentos vistos em aula:

- a) Aplique as regras da terceira forma normal (3FN) para normalizá-la.

Exercício 3

O uso dos comandos CREATE DATABASE e SHOW DATABASE são comuns nos SGBDs, explique com suas palavras qual a importância deles.

Exercício 4

Pesquise como disponibilizar um banco de dados para trabalhar com a linguagem Java com o servidor MySQL.

Exercício 5

Depois de instalado o MySQL no Microsoft Windows, como proceder para criar um novo banco de dados a partir da linha de comando?

Roteiro de Atividade

1. Introdução

Esta atividade tem como objetivo praticar os conhecimentos na biblioteca Express e de uso de banco de dados com o Sequelize.

Ao desenvolver as atividades práticas de laboratório, atente sempre em implementar código de maneira organizada, formatada e estruturada: o chamado código limpo. Mantenha seu código sempre de acordo com as *guidelines* ditas pelas boas práticas de programação. Peça orientação a seu professor mediador sempre que tiver dúvidas.

Atividade 1

Crie um servidor web com auxílio do Express a partir do que você aprendeu nas Aulas 06 que salve e obtenha as informações **de um banco de dados, com auxílio do ORM Sequelize**. O programa deve implementar uma API para cadastro e consulta de produtos, conforme detalhamento a seguir.

Esta atividade é uma extensão da atividade da aula anterior, adicionando a parte de banco de dados.

1. Os produtos devem possuir os campos nome, descrição e preço.
2. Os campos “nome” e “descrição” devem ser textuais
3. O campo preço deve ser numérico
4. O campos nome e preço são de cadastro obrigatório
5. Utilize a biblioteca ajv para validação (<https://ajv.js.org/guide/getting-started.html>)
6. As validações devem ser implementadas como um middleware do Express
7. As seguintes rotas devem estar disponíveis
 - a. GET /produtos – Lista os produtos cadastrados em formato JSON
 - b. POST /produtos – Cadastra um produto
 - c. PUT /produtos/:id – Atualiza os dados de um produto
 - d. DELETE /produtos/:id – Remove determinado produto

Atividade 2

Adicione ao cadastro de produto a possibilidade de adicionar tags. Siga os seguintes critérios:

1. Cada produto pode conter 0 ou vários tags.
2. Cada tag deverá ser um texto salvo no banco de dados em uma tabela específica.
3. Use o conceito de relacionamento para esta ação
4. Implementa uma migração para a inclusão dos campos necessários

O programa deve conseguir responder com sucesso a esse tipo de requisição:

POST http://HOST:PORTA/produto

Content-Type: application/json

```
{  
  "nome": "Nome do produto..",  
  "descricao": "Produto interessante que resolve...",  
  "preco": 90.8,  
  "tags": ["foo1", "ba1"]  
}
```