\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Você foi chamad@ para realizar a **criação de um Banco de Dados** (BD) no modelo **relacional** (que contenha entidades [tabela], atributos [colunas] e tuplas [linhas]) utilizando o Sistema Gerenciador de Banco de Dados (SGBD) **MySQL**.

Com base na **análise de requisitos** (requisito = especificações do cliente), construa o Banco de dados (BD) que contenha as seguintes informações...

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. O Banco de Dados (BD) deve registrar uma **lista de atletas** que atuam na NBA (National Basketball Association), tendo como informações mais relevantes os seguintes **atributos**: (3.50 pontos)

Nome do atleta;

Universidade ou país (e.g.: apenas uma ou um por atleta);

Remuneração;

Equipe atual;

Posição (e.g.: armador, ala-armador, ala, ala-pivô, pivô);

Especialidade (e.g.: enterrada, três pontos, rebote ofensivo, rebote defencivo, drible, et cetera)

Lesionado (e.g.: sim ou não);

Patrocínio esportivo (e.g.: Nike, Adidas, Under Armour, Não Possui, et cetera)

Com base em tais informações realize a modelagem do Banco de dados (BD) na seguinte ordem:

1º Encontrar atributos (colunas) com **dados repetitivos** [regra 15%].

2º Uso da regra de **atributos compostos** da 1º forma normal (1FN) {e.g.: nome de rua}.

3º Uso da regra de **atributos multivalorados** da 1º forma normal (1FN) {e.g.: telefone}.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Após a separação das tabelas **crie o banco no MySQL** com o nome nfl\_db e **realize o relacionamento** entre elas no MySQL. (2.00 pontos)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Adicione os dados de **ao menos uma equipe** no Banco de Dados (BD) com um **insert** (e.g.: Los Angeles Lakers). (1.00 ponto)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Salve o modelo** do diagrama **e o script** (create database, use database, insert). (1.00 ponto)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Crie ao menos **um objeto** (**Procedure** [insert apenas], **View** [select com inner join] ou **Trigger** [delete com insert no backup]). (2.50 pontos)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Procedure (com insert): Atua como uma função para insert, select, delete, update, et cetera

**DELIMITER $;**

**CREATE PROCEDURE <nome do procedure>(**

**IN <p\_atributo 1> INT,**

**IN <p\_atributo 2> VARCHAR(45),**

**IN <p\_atributo n> VARCHAR(45)**

**)**

**BEGIN**

**<commando de insert aqui>;**

**INSERT INTO <tabela>(<atrib 1>, <atrib 2>, <atrib n>)**

**VALUES(<p\_atrib 1>, <p\_atrib 2>, <p\_atrib n>);**

**END $**

**CALL <nome do procedure>(<dado 1>, <dado 2>, <dado n>);**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

View (com inner join): Minimizando um select [cria nova tabela]

**CREATE VIEW <nome da view> AS**

**SELECT <tabela 1>.<colunas 1>, <tabela 2>.<colunas 2>, <tabela n>.<colunas n>**

**FROM <tabela principal>**

**INNER JOIN <tabela anexada 1>**

**ON <tabela principal>.<coluna em comum> = <tabela anexada 1>.<coluna comum>**

**INNER JOIN <tabela anexada 2>**

**ON <tabela principal>.<coluna em comum> = <tabela anexada 2>.<coluna comum>**

**ORDER BY <ordem desejada>;**

**SELECT \* FROM <nome da view>;**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Trigger (com delete e insert):

**DELIMITER $;**

**CREATE TRIGGER <nome do trigger>**

**AFTER DELETE ON <tabela alvo>**

**FOR EACH ROW**

**BEGIN**

**INSERT INTO <tabela bkp já existente>(<atrib 1>, <atrib 2>, <atrib n>)**

**VALUES(OLD.<atrib 1 tabela>, OLD.<atrib 2 tabela>, OLD.<atrib n tabela>);**

**END**

**$**