ATIVIDADE DE BD2:

NOME: JÚLIA DE SOUZA ALVES 2DS

1.Escreva o comando SQL para **criar um banco** de dados chamado CineMarvelDC

Resultado:

CREATE DATABASE if not exists CineMarvelDC;

use CineMarvelDC ;

2. **Crie uma tabela** chamada vilao com os seguintes campos (0,5 ponto):

● idVilao (inteiro, chave primária)

● nome (texto até 100 caracteres, obrigatório)

● codinome (texto até 100 caracteres)

● origem (texto até 50 caracteres)

Resultado:

CREATE TABLE vilao (

idVilao INT AUTO\_INCREMENT PRIMARY KEY,

nome VARCHAR (100) NOT NULL,

Codinome VARCHAR (100),

origem VARCHAR(50)

);

3. **Altere a tabela** vilao para adicionar o campo nivel\_periculosidade (inteiro)

Resultado:

ALTER TABLE vilao

ADD COLUMN nivel\_periculosidade INT;

4. Escreva um comando para **criar uma tabela** confronto que contenha (0,5 ponto):

● idConfronto (chave primária)

● idHeroi e idVilao (inteiros, chaves estrangeiras referenciando tabelas heroi e vilao)

Resultado:

CREATE TABLE confronto (

idConfronto INT AUTO\_INCREMENT PRIMARY KEY,

idHeroi INT,

idvilao INT,

local VARCHAR(100) NOT NULL,

resultado VARCHAR(20),

FOREIGN KEY (idHeroi) REFERENCES heroi(idHeroi),

FOREIGN KEY (idVilao) REFERENCES vilao(idVilao)

);

5. **Insira** um vilão chamado *Arthur Fleck* com codinome *Coringa*, origem *DC*, e nível de periculosidade 10 (0,5 ponto).

Resultado:

INSERT INTO vilao (Nome, Codinome, Origem, nivel\_periculosidade)

VALUES ('Arthur Fleck', 'Coringa', 'DC', 10);

6. **Liste** todos os vilões com nível de periculosidade maior que 7

Resultado:

SELECT \* FROM vilao

WHERE nivel\_periculosidade > 7;

7. **Selecion**e os nomes dos heróis cujo codinome começa com a letra "S".

Resultado:

SELECT nome

FROM heroi

WHERE codinome LIKE 'S%';

8. **Liste** todos os confrontos ordenados por local em ordem alfabética decrescente

Resultado:

SELECT \* FROM confronto

ORDER BY local DESC;

9. **Mostre** o nome do herói, o nome do vilão e o resultado dos confrontos, unindo as tabelas heroi, vilao e confronto.

Resultado:

SELECT

heroi.nome,

vilao.nome,

resultado

FROM

confronto

JOIN heroi ON heroi.idheroi = confronto.idHeroi

JOIN vilao ON vilao.idVilao = confronto.idVilao;

10. **Liste** os nomes dos vilões que já participaram de algum confronto com o herói chamado *Capitão América*.

Resultado:

SELECT DISTINCT vilao.nome

FROM confronto

JOIN heroi ON heroi.idheroi = confronto.idHeroi

JOIN vilao ON vilao.idVilao = confronto.idVilao

WHERE heroi.nome = 'Capitão América';

11. **Quantos** vilões existem no banco de dados?

Resultado:

SELECT COUNT(\*) AS total\_viloes

FROM vilao;

12. **Calcule a média** de periculosidade dos vilões agrupados por origem.

Resultado:

SELECT

origem,

AVG(periculosidade) AS media\_periculosidade

FROM

vilao

GROUP BY

origem;

13. Liste as origens que possuem mais de 2 vilões cadastrados.

Resultado:  
SELECT

origem,

COUNT(\*) AS quantidade\_viloes

FROM

vilao

GROUP BY

origem

HAVING

COUNT(\*) > 2;