**João Monteiro-98647**

**Madalena Jorge-98666**

**Diogo Azeitona-98591**

**PROJETO BD – PARTE 2**

* PESQUISAS DE DADOS-SQL & VIEWS

1.

DROP VIEW listactipo;

CREATE VIEW listactipo AS SELECT p.Nome, p.Data\_de\_publicacao,'Livro' AS Tipo\_de\_publicacao

FROM publicacao p , Edicao\_de\_Livro el

WHERE p.Id=el.Publicacao\_Id

UNION

SELECT p.Nome, p.Data\_de\_publicacao, 'Monografia' AS Tipo\_de\_publicacao

From publicacao p, monografia m

Where p.Id = m.Publicacao\_Id

UNION

SELECT p.Nome, p.Data\_de\_publicacao, 'Periodico' AS Tipo\_de\_publicacao

From publicacao p, edicao\_de\_periodico pr

Where p.Id = pr.Publicacao\_Id;

SELECT \*

FROM listactipo

UNION

SELECT p.Nome,p.Data\_de\_publicacao,'n.a'

FROM publicacao p WHERE p.Nome NOT IN (SELECT Nome FROM listactipo)

2.

DROP VIEW maisreservas;

CREATE VIEW MaisReservas (pNome,Valor) AS

SELECT p.Nome, count(\*)

FROM publicacao p, reserva r

WHERE p.Id=r.Publicacao\_Id\_ AND r.Data\_e\_Hora<=DATE\_SUB(NOW(), INTERVAL -6 MONTH) GROUP by p.Nome;

SELECT m.pNome, m.Valor

FROM MaisReservas m

WHERE m.Valor=(SELECT MAX(maisreservas.valor)

FROM maisreservas)

GROUP BY m.pNome

ORDER BY m.pNome;

3.

SELECT p.Nome , MAX(p.relevancia) , ar.Nome

FROM Publicacao p , Area\_Tematica ar

WHERE p.Area\_Tematica\_Id=ar.Id AND p.relevancia=(SELECT MAX(p.relevancia)

FROM Publicacao

WHERE ar.Id=p.Area\_Tematica\_Id) GROUP BY ar.Nome;

4.

DROP VIEW teste;

CREATE VIEW teste (pNome,aNome,Valor) AS

SELECT p.nome,a.Nome, count(\*)

FROM publicacao p, emprestimo e, area\_tematica a

WHERE p.Id=e.Publicacao\_Id AND p.Area\_Tematica\_Id=a.Id AND (e.Data\_hora>='2021-01-01' AND e.Data\_hora<='2021-06-31')

GROUP by p.Nome

ORDER BY p.Nome;

SELECT t.pNome,t.aNome,t.Valor

FROM teste t

WHERE t.Valor=(SELECT MAX(teste.valor) FROM teste)

GROUP BY t.pNome

ORDER BY t.pNome;

5.

SELECT NumLista.Nome

FROM (SELECT u.Nome AS Nome, u.Numero AS Numero, (SELECT COUNT(\*)

FROM area\_tematica WHERE area\_tematica.Id = p.Area\_Tematica\_Id) AS Num\_areas\_por\_publicacao FROM publicacao p, utente u, livro\_em\_lista\_leitura l, area\_tematica a, edicao\_de\_livro e WHERE p.Area\_Tematica\_Id = a.Id AND u.Numero = l.Lista\_de\_leitura\_Utente\_Numero\_ AND l.Edicao\_de\_Livro\_Numero\_= e.Numero AND e.Publicacao\_Id = p.Id) NumLista GROUP BY NumLista.Numero HAVING (SUM(NumLista.Num\_areas\_por\_publicacao)/(SELECT COUNT(\*) FROM area\_tematica))>0.75

6.

SELECT p.Nome

FROM publicacao p, edicao\_de\_livro el, edicao\_de\_periodico ep, monografia m

WHERE p.Id=el.Publicacao\_Id AND p.Id=m.Publicacao\_Id OR p.Id=ep.Publicacao\_Id AND p.ID = el.Publicacao\_Id OR p.Id=ep.Publicacao\_Id AND p.Id=m.Publicacao\_Id

GROUP BY p.nome;

7.

DROP VIEW media;

CREATE VIEW media (DiferencaDias, NumPubs, DiferencaDiasMax, DiferencaDiasMin) AS SELECT SUM(p.Data\_de\_aquisicao-p.Data\_de\_publicacao), COUNT(\*), MAX(p.Data\_de\_aquisicao-p.Data\_de\_publicacao), MIN(p.Data\_de\_aquisicao-p.Data\_de\_publicacao)

FROM publicacao p WHERE p.Data\_de\_publicacao IS NOT NULL AND p.Data\_de\_aquisicao IS NOT NULL;

SELECT SUM(m.DiferencaDias)/m.NumPubs AS Media, MAX(m.DiferencaDiasMax)AS Maximo, MIN(m.DiferencaDiasMin) AS Minimo

FROM media m

GROUP BY m.DiferencaDias;

* AUTOMATISMOS: TRIGGERS, FUNCTIONS & STORED PROCEDURES

1.(Procedimento)

BEGIN

SELECT \*

FROM exemplar e

WHERE e.Publicacao\_Id=Id;

END

2.(Procedimento)

BEGIN

SELECT a.Nome, COUNT(p.Id) AS Qtd

FROM edicao\_de\_livro ed, livro\_em\_lista\_leitura ll, publicacao p, area\_tematica a

WHERE ll.Lista\_de\_leitura\_Nome\_=NomeListaLeitura AND ll.Edicao\_de\_Livro\_Livro\_Id\_=ed.Livro\_Id AND ed.Publicacao\_Id=p.Id AND p.Area\_Tematica\_Id=a.Id

GROUP BY a.nome;

END

3.(Procedimento)

BEGIN

Declare diasAtraso numeric;

Declare valorFixo numeric;

SET valorFixo=8;

IF((SELECT COUNT(\*)

FROM emprestimo

WHERE emprestimo.Data\_de\_devolucao IS NULL )>0) THEN

SELECT datediff(now(), emprestimo.Data\_de\_devolucao\_limite) INTO diasAtraso FROM emprestimo, emprestimo\_com\_multa;

UPDATE emprestimo\_com\_multa

SET emprestimo\_com\_multa.Valor\_actual\_por\_atraso =valorFixo\*diasAtraso

WHERE diasAtraso>0;

END IF;

END

4.(Procedimento)

BEGIN

DECLARE numUt numeric;

IF((SELECT COUNT(\*)

FROM emprestimo\_com\_multa ep, emprestimo e

WHERE ep.Valor\_por\_extravio>0 AND ep.Numero=e.Numero AND e.Data\_de\_devolucao\_limite <= DATE\_SUB(NOW(), INTERVAL -1 MONTH) AND e.Data\_de\_devolucao IS NULL )>0)THEN

SET numUt = (SELECT e.Utente\_Numero

FROM emprestimo\_com\_multa ep, emprestimo e

WHERE ep.Valor\_por\_extravio>0 AND ep.Numero=e.Numero AND e.Data\_de\_devolucao\_limite <= DATE\_SUB(NOW(), INTERVAL -1 MONTH) AND e.Data\_de\_devolucao IS NULL);

INSERT INTO utente\_suspenso (`Numero`, `Data\_inicio`, `Data\_fim`) VALUES (numUt, CURRENT\_DATE(), DATE\_ADD(CURDATE(), INTERVAL 1 MONTH));

END IF;

END

5.(TRIGGER – emprestimo,AFTER,INSERT)

BEGIN

IF (new.Publicacao\_Id IN (SELECT(p.Id)

FROM publicacao p, emprestimo e

WHERE p.Id=e.Publicacao\_Id)) THEN

UPDATE publicacao

SET Qtd\_Emprestimos=Qtd\_Emprestimos+1

WHERE publicacao.Id=new.Publicacao\_Id;

END IF;

END

6.(Função – INPUT NrExemplar TIPO int TAMANHO 11)

BEGIN

DECLARE retorno VARCHAR(50);

DECLARE exmnumero VARCHAR(50);

IF((SELECT COUNT(\*)

FROM exemplar

WHERE nr = NrExemplar) = 0 ) THEN

SET retorno = 'Exemplar não existe';

ELSEIF (SELECT COUNT(\*)

FROM exemplar e, publicacao p

WHERE e.Nr = NrExemplar AND p.id=e.Publicacao\_Id AND e.Pode\_ser\_emprestado = 1 AND NOT EXISTS(SELECT reserva.Data\_e\_Hora FROM reserva WHERE reserva.Data\_e\_Hora <= DATE\_ADD(NOW(), INTERVAL -1 MONTH )) =0) THEN

SET retorno = 'Exemplar Disponivel';

ELSE SET retorno ='Exemplar Indisponivel';

END IF;

RETURN (retorno);

END

7.(TRIGGER – \*Várias tabelas\*,BEFORE,INSERT)

//edicao\_de\_livro

BEGIN

IF (new.Publicacao\_Id IN (SELECT(e.Publicacao\_Id)

FROM edicao\_de\_periodico e

WHERE e.Publicacao\_Id = new.Publicacao\_Id)) THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Já existe a publicação como periódico'; ELSEIF(new.Publicacao\_Id IN (SELECT(m.Publicacao\_Id)

FROM monografia m

WHERE m.Publicacao\_Id = new.Publicacao\_Id)) THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Já existe a publicação como monografia';

END IF;

END

//edicao\_de\_periodico

BEGIN

IF (new.Publicacao\_Id IN (SELECT(e.Publicacao\_Id)

FROM edicao\_de\_livro e

WHERE e.Publicacao\_Id = new.Publicacao\_Id)) THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Já existe a publicação como livro'; ELSEIF(new.Publicacao\_Id IN (SELECT(m.Publicacao\_Id)

FROM monografia m

WHERE m.Publicacao\_Id = new.Publicacao\_Id)) THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Já existe a publicação como monografia';

END IF;

END

//monografia

BEGIN

IF (new.Publicacao\_Id IN (SELECT(e.Publicacao\_Id)

FROM edicao\_de\_periodico e

WHERE e.Publicacao\_Id = new.Publicacao\_Id)) THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Já existe a publicação como periódico'; ELSEIF(new.Publicacao\_Id IN (SELECT(el.Publicacao\_Id)

FROM edicao\_de\_livro el

WHERE el.Publicacao\_Id = new.Publicacao\_Id)) THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Já existe a publicação como livro';

END IF;

END

8. (TRIGGER – edicao\_de\_periodico,AFTER,INSERT)

BEGIN

UPDATE exemplar SET exemplar.Pode\_ser\_emprestado=1

WHERE exemplar.Pode\_ser\_emprestado=0 AND exemplar.Nr < new.Numero AND exemplar.Publicacao\_Id=new.Publicacao\_Id;

END