

## Atividade BD2

Maria Eduarda Garcia - 11796621

Mirela Mei - 11208392

```
mirelameic@pop-os -
postgres=# \d
List of relations
Schema | Name      | Type  | Owner
-----+-----+-----+-----
public | artista   | table | postgres
public | colecao   | table | postgres
public | emprestados | table | postgres
public | esculturas | table | postgres
public | exposicoes | table | postgres
public | exposto_em | table | postgres
public | objetos_arte | table | postgres
public | outros    | table | postgres
public | permanentes | table | postgres
public | pinturas  | table | postgres
(10 rows)

postgres=# |
```

1. Adicione um campo Custo nos objetos de arte e remova esse campo dos objetos permanentes.

```
ALTER TABLE OBJETOS_ARTE ADD COLUMN CUSTO DOUBLE PRECISION NOT NULL;
```

```
postgres=# ALTER TABLE OBJETOS_ARTE ADD COLUMN CUSTO DOUBLE PRECISION NOT NULL;
ALTER TABLE
postgres=# \d objetos_arte
Table "public.objetos_arte"
Column      | Type          | Collation | Nullable | Default
-----+-----+-----+-----+-----
num_id      | integer       |           | not null |
titulo      | character varying(100) |           | not null |
nome_artista | character varying(200) |           | not null |
descricao   | character varying(200) |           | not null |
ano_criacao | date          |           |          |
periodo_art | character varying(100) |           | not null |
pais_cultura | character varying(100) |           | not null |
estilo      | character varying(100) |           | not null |
tipo        | tipo          |           | not null |
status      | status        |           | not null |
custo       | double precision |           | not null |
Indexes:
    "objetos_arte_pkey" PRIMARY KEY, btree (num_id)
Foreign-key constraints:
    "objetosarte_artista_fk" FOREIGN KEY (nome_artista) REFERENCES artista(nome)
Referenced by:
    TABLE "emprestados" CONSTRAINT "emprestados_objarte_fk" FOREIGN KEY (num_obj4) REFERENCES objetos_arte(num_id)
    TABLE "esculturas" CONSTRAINT "esculturas_objarte_fk" FOREIGN KEY (num_obj2) REFERENCES objetos_arte(num_id)
    TABLE "exposto_em" CONSTRAINT "expostoem_objarte_fk" FOREIGN KEY (num_obj6) REFERENCES objetos_arte(num_id)
    TABLE "outros" CONSTRAINT "outros_objarte_fk" FOREIGN KEY (num_obj3) REFERENCES objetos_arte(num_id)
    TABLE "permanentes" CONSTRAINT "permanentes_objarte_fk" FOREIGN KEY (num_obj5) REFERENCES objetos_arte(num_id)
    TABLE "pinturas" CONSTRAINT "pinturas_objarte_fk" FOREIGN KEY (num_obj1) REFERENCES objetos_arte(num_id)
```

```
ALTER TABLE PERMANENTES DROP COLUMN CUSTO;
```

```
postgres=# ALTER TABLE PERMANENTES DROP COLUMN CUSTO;
ALTER TABLE
postgres=# \d permanentes
Table "public.permanentes"
Column      | Type          | Collation | Nullable | Default
-----+-----+-----+-----+-----
num_obj5    | integer       |           | not null |
em_exposicao | bit(1)        |           | not null |
data_aquisicao | date          |           | not null |
Indexes:
    "permanentes_pkey" PRIMARY KEY, btree (num_obj5)

postgres=# |
```

2. Listar os objetos de arte (Numid e título) de artistas desconhecidos.

```
SELECT NUM_ID, TITULO FROM OBJETOS_ARTE WHERE NOME_ARTISTA IS NULL;
```

```

postgres=# SELECT NUM_ID, TITULO FROM OBJETOS_ARTE WHERE NOME_ARTISTA IS NULL;
 num_id | titulo
-----+-----
      3 | Castelo
      4 | A Ponte
(2 rows)

postgres=#

```

**3. Listar os objetos de arte (Numid e título) expostos em todas as exposições de 2010.**

```

SELECT arte.NUM_ID, arte.TITULO FROM OBJETOS_ARTE arte
JOIN EXPOSTO_EM exposto
  ON arte.NUM_ID = exposto.NUM_OBJ6
JOIN EXPOSICOES exposicao
  ON exposto.NOME_EXPOSICAO = exposicao.NOME_EXPOSICAO
WHERE exposicao.DATA_INICIO = DATE '2010-01-01';

```

```

postgres=# SELECT arte.NUM_ID, arte.TITULO FROM OBJETOS_ARTE arte
JOIN EXPOSTO_EM exposto
  ON arte.NUM_ID = exposto.NUM_OBJ6
JOIN EXPOSICOES exposicao
  ON exposto.NOME_EXPOSICAO = exposicao.NOME_EXPOSICAO
WHERE exposicao.DATA_INICIO = DATE '2010-01-01';
 num_id | titulo
-----+-----
      1 | Abaporu
      2 | A Negra
      3 | Castelo
      4 | A Ponte
(4 rows)

```

**4. Listar os objetos de arte (Numid, título, coleção) criados pelo autor X (Nome).**

```

SELECT arte.NUM_ID, arte.TITULO, col.NOME_COLECAO
FROM OBJETOS_ARTE arte
FULL JOIN PERMANENTES per
  ON arte.NUM_ID = per.NUM_OBJ5
FULL JOIN EMPRESTADOS emp
  ON arte.NUM_ID = emp.NUM_OBJ4
FULL JOIN COLECAO col
  ON per.COL_PERMANENTE = col.NOME_COLECAO OR emp.NOME_COLECAO
= col.NOME_COLECAO
WHERE NOME_ARTISTA = 'Vicent Van Gogh';

```

```

postgres=# SELECT arte.NUM_ID, arte.TITULO, col.NOME_COLECAO
postgres=# FROM OBJETOS_ARTE arte
postgres=# FULL JOIN PERMANENTES per
postgres=#
postgres=# ON arte.NUM_ID = per.NUM_OBJ5
postgres=# FULL JOIN EMPRESTADOS emp
postgres=#
postgres=# ON arte.NUM_ID = emp.NUM_OBJ4
postgres=# FULL JOIN COLECAO col
postgres=#
postgres=# ON per.COL_PERMANENTE = col.NOME_COLECAO OR emp.NOME_COLECAO = col.NOME_COLECAO
postgres=# WHERE NOME_ARTISTA = 'Vicent Van Gogh';
 num_id | titulo | nome_colecao
-----+-----+-----
      6 | A Noite Estrelada | Colecao Permanente
      7 | Os Girassóis | Colecao Permanente
      8 | Auto-Retrato | Colecao de Inverno
(3 rows)

```

**5. Listar as coleções (Nome e tipo) que contêm objetos de arte produzidos pelo artista X.**

```

SELECT col.NOME_COLECAO, col.TIPO_COLECAO

```

```

FROM OBJETOS_ARTE arte
FULL JOIN PERMANENTES per
    ON arte.NUM_ID = per.NUM_OBJ5
FULL JOIN EMPRESTADOS emp
    ON arte.NUM_ID = emp.NUM_OBJ4
FULL JOIN COLECAO col
    ON per.COL_PERMANENTE = col.NOME_COLECAO OR emp.NOME_COLECAO
= col.NOME_COLECAO
WHERE NOME_ARTISTA = 'Vicent Van Gogh';

```

```

postgres=# SELECT col.NOME_COLECAO, col.TIPO_COLECAO
postgres=# FROM OBJETOS_ARTE arte
postgres=# FULL JOIN PERMANENTES per
postgres=#
postgres=# ON arte.NUM_ID = per.NUM_OBJ5
postgres=# FULL JOIN EMPRESTADOS emp
postgres=#
postgres=# ON arte.NUM_ID = emp.NUM_OBJ4
postgres=# FULL JOIN COLECAO col
postgres=#
postgres=# ON per.COL_PERMANENTE = col.NOME_COLECAO OR emp.NOME_COLECAO = col.NOME_COLECAO
postgres=# WHERE NOME_ARTISTA = 'Vicent Van Gogh';
   nome_colecao   | tipo_colecao
-----+-----
 Colecao Permanente | Tipo
 Colecao Permanente | Tipo
 Colecao de Inverno | Tipo
(3 rows)

```

6. Qual é o custo total da coleção permanente do museu?

```

SELECT SUM(CUSTO) AS CUSTO_TOTAL FROM OBJETOS_ARTE
WHERE STATUS = 'PERMANENTE';

```

```

postgres=# SELECT SUM(CUSTO) AS CUSTO_TOTAL FROM OBJETOS_ARTE
postgres=# WHERE STATUS = 'PERMANENTE';
   custo_total
-----
    1281000
(1 row)

```

7. Quais são os objetos (Numid e título) da coleção permanente cujo custo é maior que o custo médio de todos os objetos do museu?

```

SELECT NUM_ID, TITULO FROM OBJETOS_ARTE
WHERE STATUS='PERMANENTE' AND CUSTO>
    (SELECT AVG(CUSTO) FROM OBJETOS_ARTE);

```

```

postgres=# SELECT NUM_ID, TITULO FROM OBJETOS_ARTE
postgres=# WHERE STATUS='PERMANENTE' AND CUSTO>
postgres=# (SELECT AVG(CUSTO) FROM OBJETOS_ARTE);
 num_id |          titulo
-----+-----
      10 | A Persistência da Memória
(1 row)

```

8. Quais são as coleções (sem considerar a permanente) com o maior número de objetos de arte emprestados?

```

SELECT NOME_COLECAO FROM EMPRESTADOS
GROUP BY NOME_COLECAO
HAVING (COUNT(EMPRESTADOS)=COUNT(NUM_OBJ4));

```

```

postgres=# SELECT NOME_COLECAO FROM EMPRESTADOS
postgres=# GROUP BY NOME_COLECAO
postgres=# HAVING (COUNT(EMPRESTADOS)=COUNT(NUM_OBJ4));
 nome_colecao
-----
 Colecao de Verao
 Colecao de Inverno
(2 rows)

```

**9. Quais são as coleções (sem considerar a permanente) com o maior custo total de objetos emprestados?**

```

SELECT NOME_COLECAO FROM EMPRESTADOS, OBJETOS_ARTE
WHERE STATUS='EMPRESTADO'
GROUP BY NOME_COLECAO
HAVING (SUM(OBJETOS_ARTE.CUSTO)>AVG(OBJETOS_ARTE.CUSTO));

```

```

postgres=# SELECT NOME_COLECAO FROM EMPRESTADOS, OBJETOS_ARTE
postgres=# WHERE STATUS='EMPRESTADO'
postgres=# GROUP BY NOME_COLECAO
postgres=# HAVING (SUM(OBJETOS_ARTE.CUSTO)>AVG(OBJETOS_ARTE.CUSTO));
 nome_colecao
-----
 Colecao de Verao
 Colecao de Inverno
(2 rows)

```

**10. Quais são as coleções com maior número de objetos emprestados que o número de objetos da coleção permanente?**

```

SELECT NOME_COLECAO FROM EMPRESTADOS, PERMANENTES,
OBJETOS_ARTE
WHERE STATUS='EMPRESTADO'
GROUP BY NOME_COLECAO
HAVING (SUM(NUM_OBJ4)>SUM(NUM_OBJ5));

```

```

postgres=# SELECT NOME_COLECAO FROM EMPRESTADOS, PERMANENTES, OBJETOS_ARTE
postgres=# WHERE STATUS='EMPRESTADO'
postgres=# GROUP BY NOME_COLECAO
postgres=# HAVING (SUM(NUM_OBJ4)>SUM(NUM_OBJ5));
 nome_colecao
-----
(0 rows)

```

**11. Quais são as coleções que tem contribuído em todas as exposições de 2010?**

```

SELECT NOME_COLECAO FROM EMPRESTADOS, PERMANENTES, EXPOSICOES
GROUP BY NOME_COLECAO, DATA_INICIO
HAVING (DATA_INICIO = DATE '2010-01-01');

```

```

postgres=# SELECT NOME_COLECAO FROM EMPRESTADOS, PERMANENTES, EXPOSICOES
postgres=# GROUP BY NOME_COLECAO, DATA_INICIO
postgres=# HAVING (DATA_INICIO = DATE '2010-01-01');
 nome_colecao
-----
 Colecao de Verao
 Colecao de Inverno
(2 rows)

```

**12. Remover todos os objetos emprestados da coleção Y.**

```

DELETE FROM OBJETOS_ARTE USING EMPRESTADOS

```

```
WHERE STATUS = 'EMPRESTADO' AND NOME_COLECAO = 'Colecao de Verao';
```

```
postgres=# DELETE FROM OBJETOS_ARTE USING EMPRESTADOS
postgres=# WHERE STATUS = 'EMPRESTADO' AND NOME_COLECAO = 'Colecao de Verao';
ERROR:  update or delete on table "objetos_arte" violates foreign key constraint "pinturas_objarte_fk" on table "pinturas"
DETAIL:  Key (num_id)=(1) is still referenced from table "pinturas".
```

**13. Transferir o objeto de título X da coleção Y para coleção permanente.**

```
UPDATE OBJETOS_ARTE
SET
    STATUS = 'PERMANENTE',
    NOME_COLECAO = 'Colecao Permanente'
FROM
    OBJETOS_ARTE JOIN EMPRESTADOS
    ON NUM_ID = NUM_OBJ4
WHERE
    TITULO = 'A Noite Estrelada';
```