

PARQUE DE DIVERSÕES

PROJETO – GDI

Amanda Cristina Fernandes Medeiros de Lima

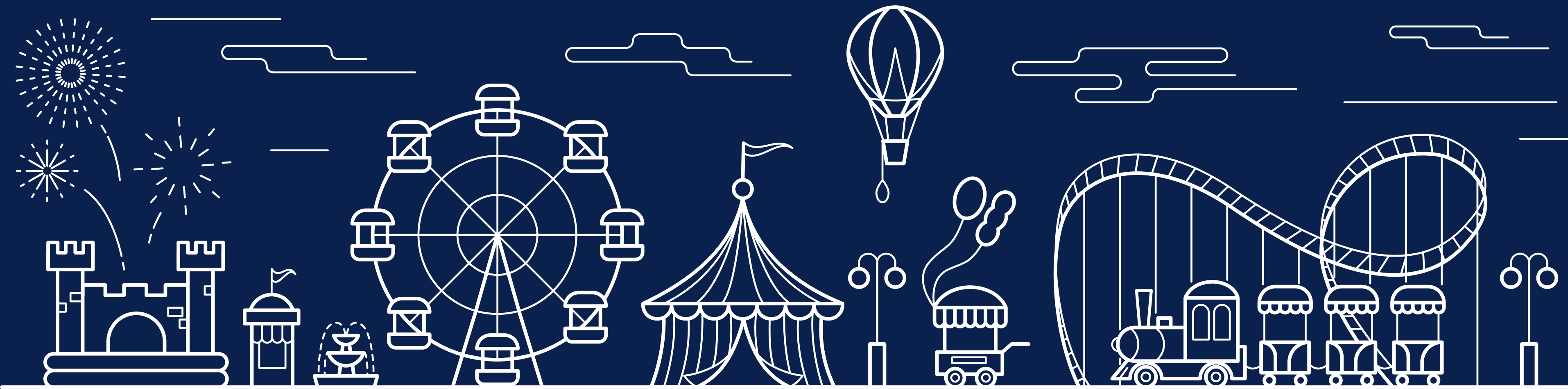
Artur Luís Vieira de Melo Boa Viagem

Caio Vinícius de Meneses Fernandes Coêlho

Maria Letícia do Nascimento Gaspar

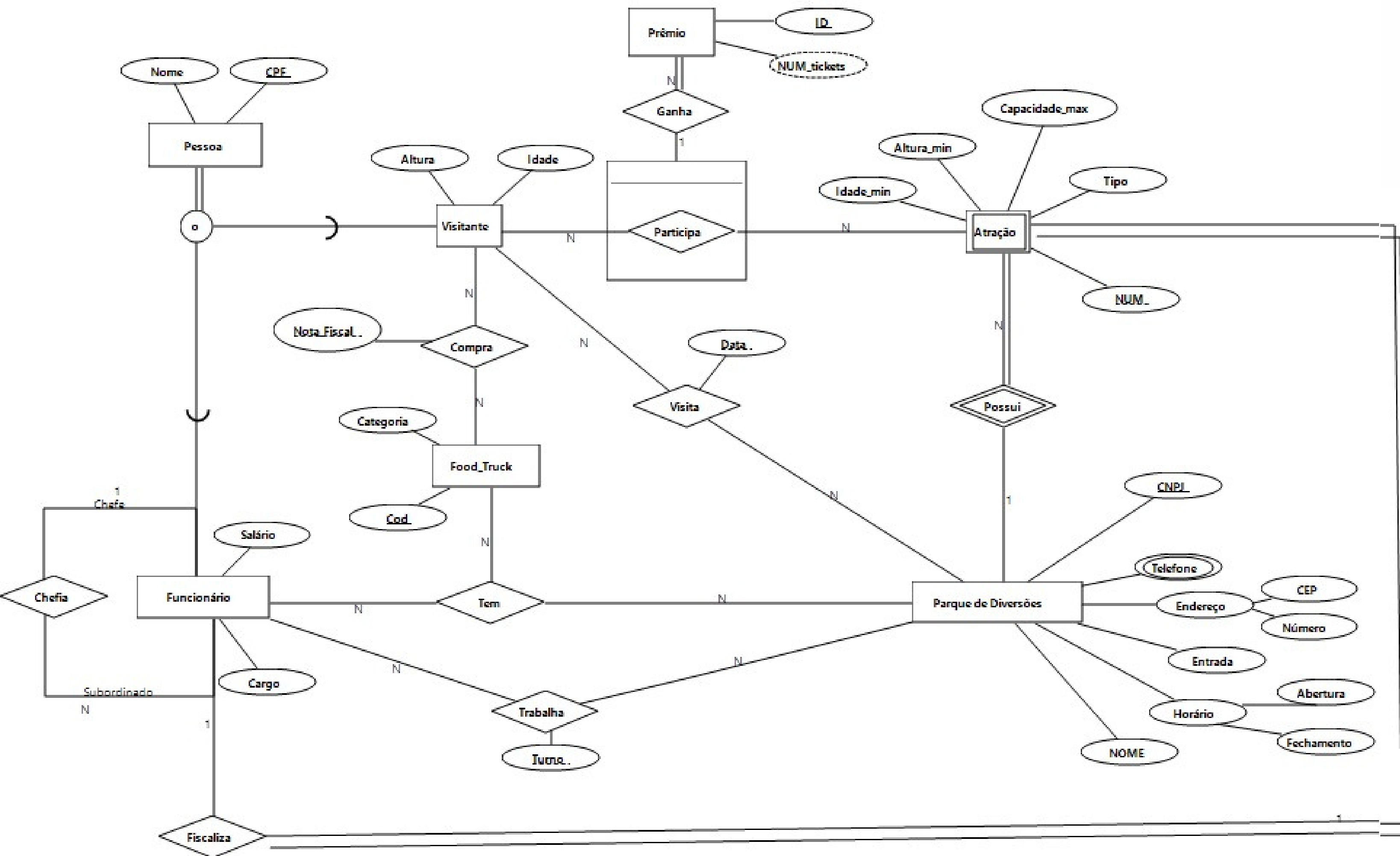
Roberto Oliveira Carneiro de Albuquerque

Victória Barbosa Cesar Figueiredo



MODELO CONCEITUAL





MODELO LÓGICO



FUNCIONARIO (CPF, NOME, SALARIO, CARGO, CPF_CHEFE)

CPF_CHEFE → FUNCIONARIO(CPF)

VISITANTE (CPF, NOME, IDADE, ALTURA)

PARQUE (CNPJ, NOME, END_CEP, END_NUM, ENTRADA, HOR_ABERTURA,
HOR_FECHAMENTO, TELEFONE1, TELEFONE2)

TRABALHA(CPF_FUNC, CNPJ, TURNO)

CPF_FUNC → FUNCIONARIO(CPF)

CNPJ → PARQUE(CNPJ)

ATRACAO (CNPJ, NUM, TIPO, ALTURA_MIN, IDADE_MIN, CAPACIDADE_MAX, TIPO,
[CPF_FUNC]!)

CNPJ → PARQUE(CNPJ)

CPF_FUNC → FUNCIONARIO(CPF)

FOOD_TRUCK (COD, CATEGORIA)

VISITA (CPF_VISITANTE, CNPJ, DATA)

CPF_VISITANTE → VISITANTE (CPF)

CNPJ → PARQUE_DE_DIVERSOES (CNPJ)

TEM (CPF_FUNC, CNPJ, COD)

CPF_FUNC → FUNCIONARIO (CPF)

CNPJ → PARQUE_DE_DIVERSOES (CNPJ)

COD → FOOD_TRUCK (COD)

PARTICIPA (CPF_VISITANTE, CNPJ, NUM)

CPF_VISITANTE → VISITANTE (CNPJ)

(CNPJ, NUM) → ATRACAO (CNPJ, NUM)

PREMIO (ID, NUM_TICKETS, CPF_VISITANTE!, CNPJ!, NUM!)

(CPF_VISITANTE, CNPJ, NUM) → PARTICIPA (CPF_VISITANTE, CNPJ, NUM)

COMPRA (CPF_VISITANTE, NOTA_FISCAL, COD)

CPF_VISITANTE → VISITANTE(CPF)

COD → FOOD_TRUCK(COD)

CONSULTAS



Consultas com group by/having

```
-- Projetar os CPFs dos funcionários que trabalham em mais de 1 parque
SELECT T.CPF_FUNC, COUNT(*) AS QTD_PARQUE
FROM TRABALHA T
GROUP BY T.CPF_FUNC
HAVING COUNT(*) > 1;
```

CPF_FUNC	QTD_PARQUE
1010	2
1011	2
1022	2
1111	2
2210	2
2211	2
2212	2
2222	2
2223	2
2224	2
3333	2
9911	2
9999	2

Consultas com group by/having


```
-- Projetar as datas que houveram mais de 1 visitante no parque de CNPJ 1113
SELECT V.DATA, COUNT(*) AS QTD_VISITANTE
FROM VISITA V
WHERE V.CNPJ = '1113'
GROUP BY V.DATA
HAVING COUNT(*) > 1
```



DATA	QUANTIDADE
09-OCT-22	2
10-JUN-22	2

Junção Interna

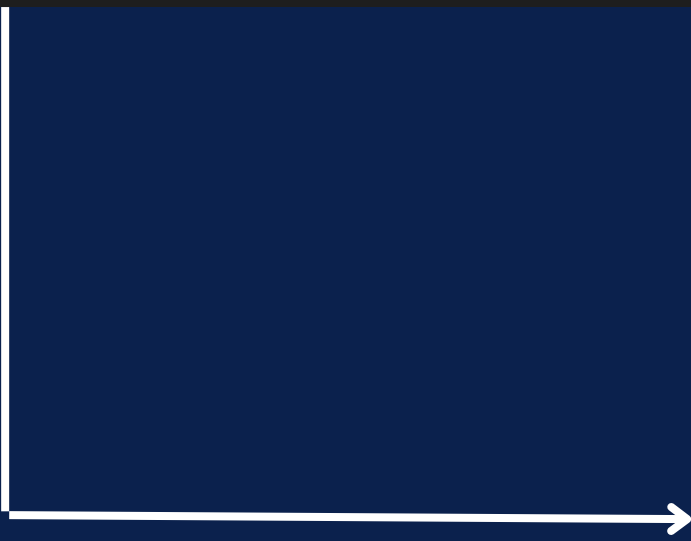
```
-- Projetar por categoria os food truck que venderam mais de 1 item  
SELECT F.CATEGORIA, COUNT(*) AS Itens_comprados  
FROM FOOD_TRUCK F INNER JOIN COMPRA C ON (F.COD = C.COD)  
GROUP BY F.CATEGORIA  
HAVING COUNT(*) > 1;
```



CATEGORIA	ITENS_COMPRADOS
PASTEL	2
ESPETINHO	4
PIZZA	2

Junção Interna

```
1 --Projeta nome e idade de todos os visitantes menores de idade no parque de cnpj 1113
2 v SELECT NOME, IDADE
3 FROM VISITANTE v INNER JOIN VISITA vi
4 ON vi.CPF_VISITANTE = v.CPF
5 WHERE v.IDADE < 18 AND vi.CNPJ = '1113'
```



NOME	IDADE
Matheus	5
Larissa	14
Bianca	11
Rebeca	7
Vinícius	8
Thiago	17
Rafael	14

Junção Externa

```
-- Projeter os nomes dos visitantes que não compraram em nenhum food_truck  
SELECT V.NOME  
FROM VISITANTE V LEFT OUTER JOIN COMPRA C ON (V.CPF = C.CPF_VISITANTE)  
WHERE C.CPF_VISITANTE IS NULL;
```

NOME
Laís
Bianca
Rebeca


Semi Junção

```
-- Projetar o nome e as idades dos visitantes que participaram de atrações  
SELECT V.NOME, V.IDADE  
FROM VISITANTE V  
WHERE EXISTS (SELECT *  
              FROM PARTICIPA P  
              WHERE V.CPF = P.CPF_VISITANTE);
```

NOME	IDADE
Matheus	5
Leonardo	36
Bianca	11
Rebeca	7
Ênio	49
Vinícius	8
Thiago	17
Rafael	14

Anti-Junção


```
-- Projetar as categoria dos food_trucks que não venderam nenhum produto
SELECT F.CATEGORIA
FROM FOOD_TRUCK F
WHERE NOT EXISTS (SELECT *
                  FROM COMPRA C
                  WHERE C.COD = F.COD);
```



CATEGORIA
CACHORRO QUENTE
CHURROS
SORVETERIA
CREPE

Subconsulta do tipo escalar


```
-- Projetar o nome do parque mais visitado
SELECT P.NOME, COUNT(*)
FROM VISITA V INNER JOIN PARQUE P ON (V.CNPJ = P.CNPJ)
GROUP BY P.NOME
HAVING COUNT(*) = (SELECT MAX(QTD_VISITANTES)
                   FROM (SELECT COUNT(*) AS QTD_VISITANTES
                         FROM VISITA V1
                         GROUP BY V1.CNPJ));
```



NOME	COUNT(*)
CIntilândia	11

Subconsulta do tipo escalar


```
-- Projeter os CPFs, nomes e alturas dos visitantes que tem altura maior que a média dos visitantes
SELECT V.CPF, V.NOME, V.ALTURA
FROM VISITANTE V
WHERE V.ALTURA > (SELECT AVG(V1.ALTURA)
                  FROM VISITANTE V1);
```



CPF	NOME	ALTURA
3344	Ênio	1.6
7788	Thiago	1.8
1112	Péricles	1.9
1516	Leonardo	1.72
1920	Laís	1.66

Subconsulta do tipo linha

```
-- listar os nomes e os cpfs do visitantes que tenham  
-- a mesma idade e a mesma altura do visitante com cpf = '1718'  
SELECT V1.NOME, V1.CPF  
FROM VISITANTE V1  
WHERE (V1.IDADE, V1.ALTURA) = (SELECT V2.IDADE, V2.ALTURA  
                                FROM VISITANTE V2  
                                WHERE V2.CPF = '1718');
```



NOME	CPF
Rafael	9910
Larissa	1718

Subconsulta do tipo tabela

```
-- Projeter os nomes dos seguranças que trabalham em apenas 1 parque
SELECT F.NOME
FROM FUNCIONARIO F
WHERE F.CPF IN (SELECT T.CPF_FUNC
                FROM TRABALHA T
                GROUP BY T.CPF_FUNC
                HAVING COUNT(*) = 1) AND F.CARGO = "SEGURANCA";
```

NOME
Mariana
João

Operação de Conjuntos

```
-- Projetar os nomes dos visitantes com mais de 15 anos que
-- compraram em algum food_truck
-- e participaram de atração
SELECT V.NOME
FROM VISITANTE V
WHERE V.IDADE > 15 AND V.CPF IN (SELECT C.CPF_VISITANTE
                                FROM COMPRA C)

INTERSECT

SELECT V.NOME
FROM VISITANTE V
WHERE V.CPF IN (SELECT P.CPF_VISITANTE
               FROM PARTICIPA P)
```

NOME
Leonardo
Thiago
Ênio

Procedimento com SQL embutida e parâmetro

```
-- PROJETAR O NOME DE TODOS OS VISITANTE QUE FORAM AO PARQUE '1113' NO ANO DE 2022
CREATE OR REPLACE PROCEDURE VISITANTES_POR_ANO (ANO VARCHAR, CNPJ VARCHAR) IS
  CURSOR CUR_VISITANTES_PARQUE_NOME IS
  SELECT DISTINCT V.NOME
  FROM VISITANTE V INNER JOIN VISITA V2 ON (V.CPF = V2.CPF_VISITANTE)
  WHERE EXTRACT(YEAR FROM (V2.DATA)) = ANO AND V2.CNPJ = CNPJ;
BEGIN
  DBMS_OUTPUT.PUT_LINE('Visitantes que foram ao parque ' || CNPJ || ' no ano de ' || ANO);
  FOR VISIT IN CUR_VISITANTES_PARQUE_NOME LOOP
    DBMS_OUTPUT.PUT_LINE(VISIT.NOME);
  END LOOP;
END;
```

```
Visitantes que foram ao parque 1113 no ano de 2022
Júlia
Ênio
Péricles
Leonardo
Larissa
Vinícius
Rebeca
Laís
Matheus
Bianca
Thiago
Rafael
```

Função com SQL embutida e parâmetro

```
-- Projetar nome do funcionário que fiscaliza determinada atração
CREATE OR REPLACE FUNCTION FUNC_FISCALIZA(TIPO_ATRACAO IN VARCHAR, PARQUE IN VARCHAR) RETURN VARCHAR IS
    NOME_FUNCIONARIO VARCHAR(30);
BEGIN
    SELECT F.NOME INTO NOME_FUNCIONARIO
    FROM FUNCIONARIO F
    WHERE F.CPF = (SELECT A.CPF_FUNC
                   FROM ATRACAO A
                   WHERE A.TIPO = TIPO_ATRACAO AND A.CNPJ = PARQUE);
    RETURN NOME_FUNCIONARIO;
END;
```

FISCAL_ATRACAO

Arthur