

FIAP



## AULA 10

SUBQUERY, SEQUENCE

INDEX, VIEW e TRIGGER

Welcome to the next evolution in higher education.

DISCIPLINA  
A **BUILDING RELATIONAL  
DATABASE**

📄 **PROFº DRº FRANCISCO D. L.**  
✉ **ABREU**  
PROFFRANCISCO@FIAP.COM.BR

# I SUMÁRIO

- ☐ Sobre VIEW
- ☐ Sobre SubQuery
- ☐ Sobre SEQUENCE

## **| OBJETIVO**

**Introduzir conceitos iniciais do SQL**

**Aplicar os conceitos no Oracle SQL Developer**

# **SOBRE VIEW**

---

- É uma representação virtual de uma tabela, que é derivada de uma ou mais tabelas existentes em um banco de dados.
- Permite que você crie consultas SQL complexas, resumos ou segmentações de dados, mas sem armazenar fisicamente os resultados
- As **VIEWS** oferecem várias vantagens:
  - **Simplificação das consultas:** permite que defina consultas complexas uma vez e as reutilize sempre que necessário
  - **Desempenho:** podem melhorar o desempenho, uma vez que o banco de dados pode armazenar em cache os resultados de consultas frequentes em vez de recalculá-los repetidamente. Segurança
  - **Segurança:** pode conceder permissões específicas de consulta em uma VIEW, ocultando detalhes sensíveis dos dados subjacentes.

## SINTAXE

- O comando **CREATE VIEW** é utilizado para criar uma visão
- Veja o exemplo da Sintaxe:

```
CREATE VIEW nome_view AS  
SELECT * FROM nome_tabela  
WHERE <condicao_logica>;
```

# I EXEMPLO DE VIEW

FIAP

## COMANDO

## RESULTADO

	NOME_CLIENTE	DS_PRODUTO	VL_PEDIDO
1	João	Celular	1100
2	João	TV	5000
3	Ana	Geladeira	3000

```
CREATE VIEW PedidosSaoPaulo AS
SELECT nome nome_cliente, ds_produto, vl_pedido
FROM CLIENTES C INNER JOIN PEDIDOS P ON C.id=P.cliente_id
WHERE Cidade = 'São Paulo';

-- consultando a VIEW
SELECT * FROM PedidosSaoPaulo;
```



# **SOBRE SUBQUERY**

---

- Também conhecida como subconsulta ou consulta aninhada, é uma consulta SQL que é incorporada dentro de outra consulta SQL
- **SUBQUERY** é usada para recuperar dados que serão usados na consulta principal para realizar operações mais complexas ou filtrar resultados
- As **SUBQUERY** oferecem várias vantagens:
  - **Filtragem de dados:** Usar o resultado de uma subquery para filtrar os resultados da consulta principal com base em alguma condição.
  - **Comparações:** Comparar valores da consulta principal com resultados de subqueries para determinar se um registro deve ser incluído.
  - **Recuperação de dados agregados:** Calcular valores agregados (como soma, média, máximo, mínimo) de subconjuntos de dados para uso na consulta principal.

# I EXEMPLO DE SUBQUERY

FIAP

## COMANDO

## RESULTADO

encontrar todos os clientes que fizeram pedidos superior a 3 mil

	NOME
1	João
2	Maria
3	Carlos



```
SELECT nome
FROM Clientes
WHERE id IN (SELECT cliente_id FROM Pedidos WHERE vl_pedido >
3000);
```

# I EXEMPLO DE SUBQUERY

FIAP

## COMANDO

## RESULTADO

encontrar a cidade com o maior número de clientes

	CIDADE	TOTAL_CLIENTES
1	São Paulo	2
2	Rio de Janeiro	2



```
SELECT C.cidade, COUNT(*) as total_clientes
FROM Clientes C
GROUP BY C.cidade
HAVING COUNT(*) = (SELECT MAX(clientes_count) FROM (SELECT
COUNT(*) as clientes_count FROM Clientes GROUP BY cidade)
subquery);
```

# SOBRE SEQUENCE

---

- Uma **SEQUENCE** é um objeto que gera valores sequenciais exclusivos.
- São frequentemente usadas para gerar valores de chave primária em tabelas, garantindo que cada registro tenha um valor único e crescente
- As **SEQUENCE** oferecem várias vantagens:
  - **Valores Únicos:** Uma sequence gera valores únicos, garantindo que cada valor gerado seja diferente dos valores anteriores.
  - **Valores Crescentes ou Decrescentes:** Sequences podem gerar valores em ordem crescente ou decrescente, dependendo de como são configuradas.
  - **Independentes de Transações:** Sequences são frequentemente independentes de transações, o que significa que os valores gerados por uma sequence não são afetados por transações concorrentes em execução no banco de dados.

## SINTAXE

- O comando **CREATE SEQUENCE** é utilizado para criar uma sequência
- Neste exemplo, estamos criando uma sequence chamada "minha\_sequence" que começa em 1, com um incremento de 1, um valor mínimo de 1, um valor máximo de 1000 e a opção "CYCLE" que faz com que a sequence recomece a partir do valor mínimo após atingir o valor máximo.

```
CREATE SEQUENCE minha_sequence  
    START WITH 1  
    INCREMENT BY 1  
    MINVALUE 1  
    MAXVALUE 1000  
    CYCLE;
```

# I EXEMPLO DE SEQUENCE

FIAP

## COMANDO

## RESULTADO

Para obter o próximo valor da sequence, você pode usar a função NEXTVAL:



```
SELECT minha_sequence.NEXTVAL FROM dual;
```



# **SOBRE INDEX**

---

- Um índice (ou "index" em inglês) é uma estrutura de dados que melhora a velocidade de recuperação de registros de uma tabela
- Os índices são usados para acelerar consultas, tornando a busca de informações mais eficiente, principalmente em tabelas grandes
- Os **INDEX** oferecem várias vantagens:
  - **Estruturar Dados** : Um índice é uma estrutura de dados que armazena uma lista ordenada de valores de uma ou mais colunas da tabela. Cada valor no índice está associado a um ou mais registros na tabela e uma referência para sua localização
  - **Consultas Eficientes**: Permite que localize registros com mais rapidez, em vez de pesquisar em toda a tabela;

## SINTAXE

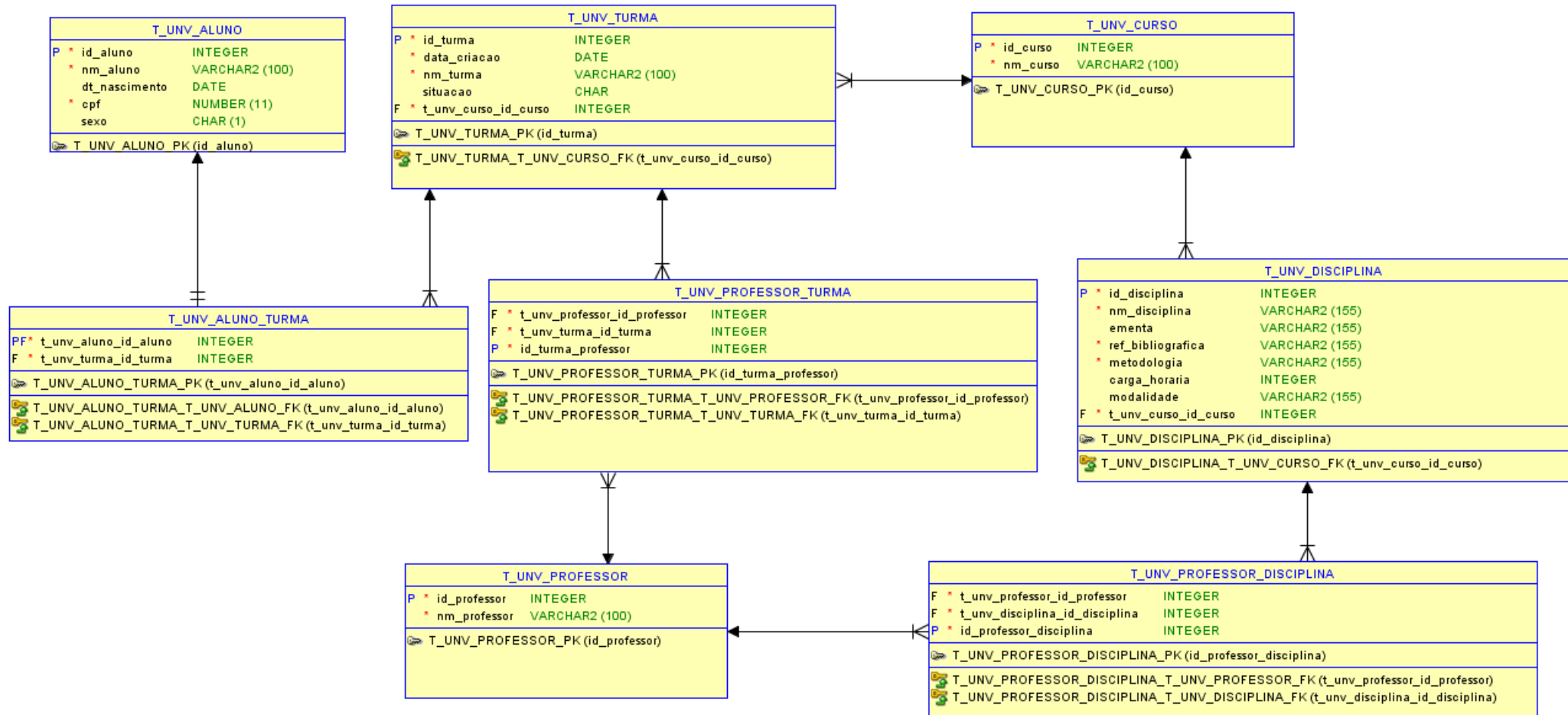
- O comando **CREATE INDEX** é utilizado para criar um índice
- Veja o Exemplo da Sintaxe

```
CREATE INDEX nome_do_indice  
ON nome_da_tabela (nome_da_coluna);
```

# EXERCÍCIO PRÁTICO

---

# EXERCÍCIO PRÁTICO



- Altere as estruturas das tabelas no SQL Developer:
  - Adicione o campo SITUACAO (CHAR(1)) na tabela ALUNO e PROFESSOR;
  - Altere o nome do campo CARGA HORARIA para CH da tabela DISCIPLINA ;
  - Altere o nome da tabela ALUNO para DISCENTE e PROFESSOR para DOCENTE;
  - Altere o tipo das colunas EMENTA, REF BIBLIOGRAFICA E METODOLOGIA para LONG VARCHAR;
  - Remova todas as tabelas existentes (Observe que há uma ordem correta de exclusão);

**Copyright © 2023 Profº Drº Francisco Douglas Lima Abreu**

Todos direitos reservados. Reprodução ou divulgação total ou parcial deste documento é expressamente proibido sem o consentimento formal, por escrito ao autor

FIAP

THE WAY WE ARE