



Faculdade
IMPACTA
TECNOLOGIA



Linguagem SQL / Banco de Dados

Aula Extra – Revisão:

Gustavo Bianchi Maia
gustavo.maia@faculdadeimpacta.com.br



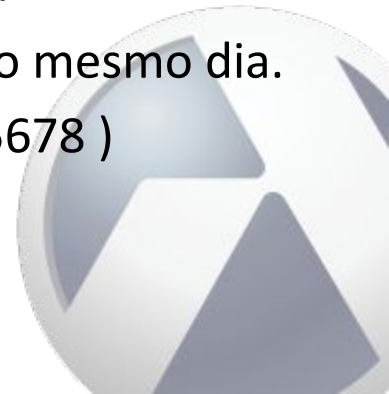


- Rever os conceitos de:
 - Modelagem
 - Construção do DER
 - Mapeamento para o modelo lógico relacional
 - Construção do modelo físico
 - Implementação das estruturas no SQL Server
 - Utilização de regras / constraints
 - Inclusão / alteração de dados (teste de carga)
 - Relatórios básicos





- Você foi contratado pela Anatel para criar um modelo de banco de dados que regule a portabilidade de celulares entre operadoras.
- Para tal, seu modelo deve ser capaz de responder às seguintes perguntas:
 - Qual a operadora atual do celular X ?
 - Qual a operadora do celular X na data YYYY/MM/DD ?
 - Quais as portabilidades (de – para) realizadas pelo celular X desde YYYY/MM/DD ?
 - Quais os celulares atuais da operadora Y ?
- Algumas regras foram estabelecidas para este modelo:
 - Nenhum celular pode estar em duas operadoras diferentes no mesmo dia.
 - Formato dos Celulares DDI + DDD + Número (ex: 5511912345678)



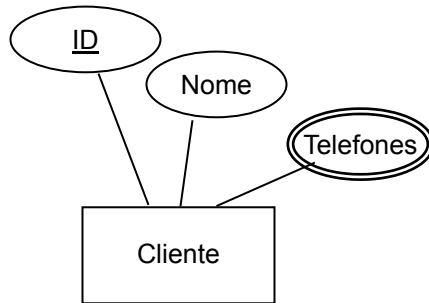


- Agora, em 10 mins:
 - Implemente o DER deste modelo
 - Implemente o modelo lógico relacional do mesmo.





- Você foi por este caminho ?



Telefone	Cliente
Id_Cliente Numero	ID Nome

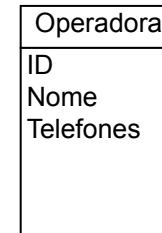
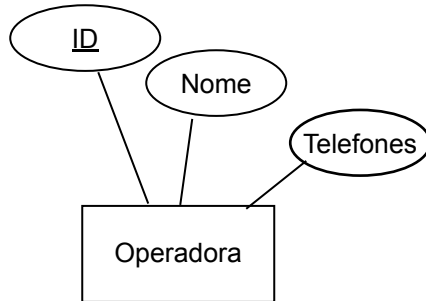
- Você falhou porque:
 - Não foi solicitado o controle de quem é o dono do celular

Lembre-se, se atenha ao mini mundo necessário para completar a tarefa, não procure soluções genéricas ou completas sem necessidade.



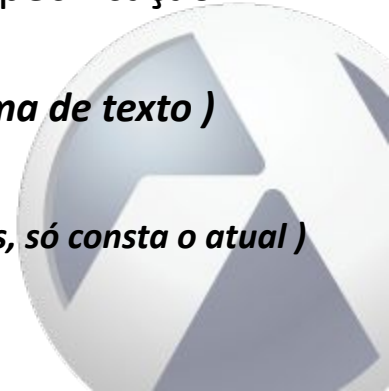


- Você foi por este caminho ?



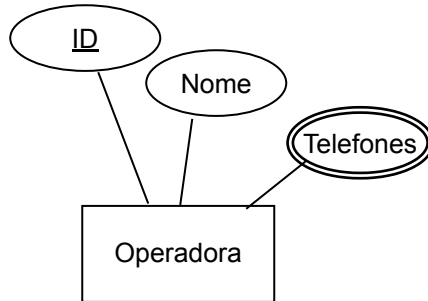
Operadora		
Id	Nome	Telefones
1	VIVO	5511912345678, 5511987654321, ...
2	TIM	5511912345678, 5511987654321, ...

- Você falhou porque:
 - Apesar de conseguir responder [de forma simplória] à pergunta / especificação :
 - Qual a operadora atual do celular X ? (***através de busca textual***)
 - Quais os celulares atuais da operadora Y ? (***um uma única lista em forma de texto***)
 - Você não responde corretamente à pergunta / especificação :
 - Qual a operadora do celular X na data YYYY/MM/DD ? (***não registra datas, só consta o atual***)





- Você foi por este caminho ?



Telefone
Id_Operadora
Numero

Operadora
ID
Nome

Telefone	
Id_operadora	Numero
1	5511912345678
1	5511987654321
2	5511912345678
2	5511987654321

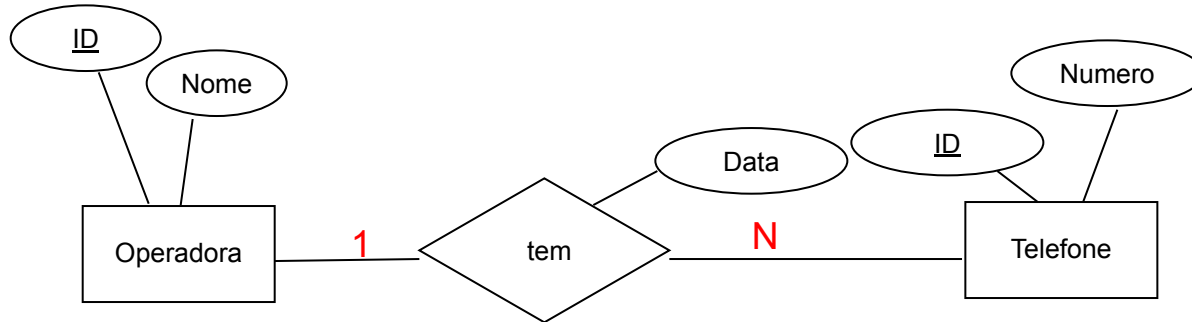
Operadora	
Id	Nome
1	VIVO
2	TIM

- Você falhou porque:
 - Apesar de conseguir responder [de forma simplória] à pergunta / especificação :
 - Qual a operadora atual do celular X ?
 - Quais os celulares atuais da operadora Y ?
 - Você não responde corretamente à pergunta / especificação :
 - Qual a operadora do celular X na data YYYY/MM/DD ? *(não registro data, só consta o atual)*





- Você foi por este caminho ?



Telefone
Id_Telefone
Numero
Id_operadora
Data

Operadora
ID
Nome

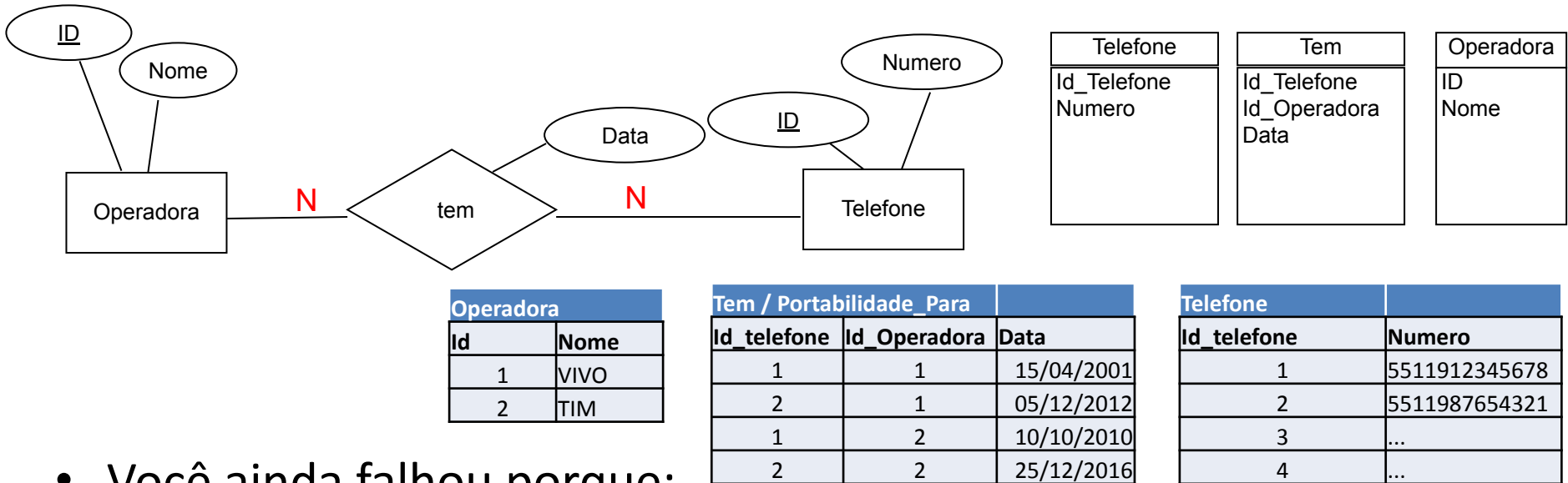
Operadora	
Id	Nome
1	VIVO
2	TIM

Telefone			
Id_Telefone	Numero	Id_operadora	Data
1	5511912345678	1	10/10/2010
2	5511987654321	1	05/12/2012
3	5511912345678	2	15/04/2001
4	5511987654321	2	25/12/2016

- Você ainda falhou porque:
 - Apesar de conseguir responder à pergunta / especificação :
 - Qual a operadora atual do celular X ?
 - Consegue garantir a unicidade de 1 celular por operadora por data
 - Você não responde corretamente à pergunta / especificação :
 - Qual a operadora do celular X na data YYYY/MM/DD ? (**Só registra a atual**)



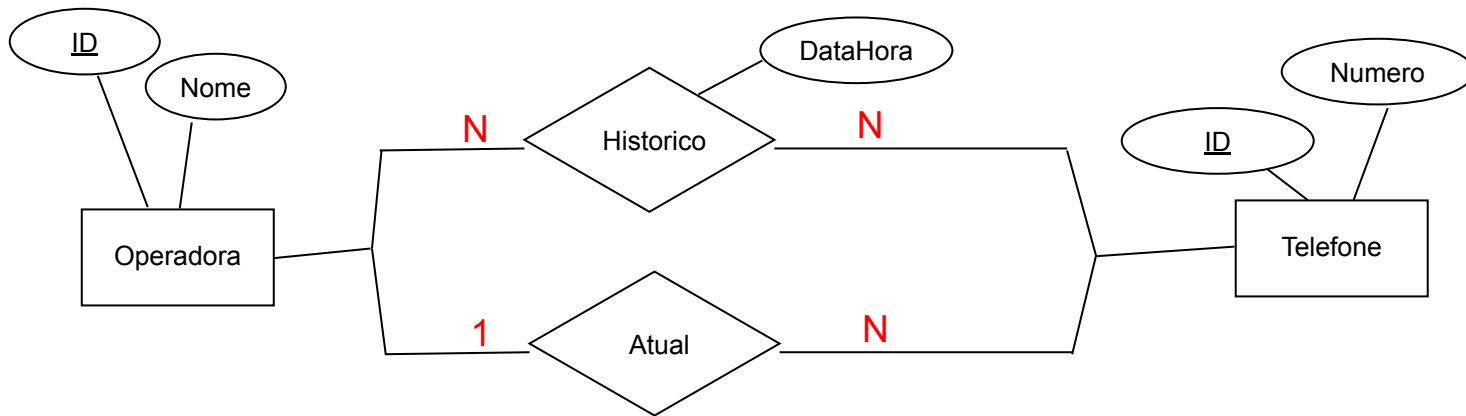
- Você foi por este caminho ?



- Você ainda falhou porque:
 - Apesar de conseguir responder à pergunta / especificação :
 - Qual a operadora do celular X na data YYYY/MM/DD ?
 - Você não responder corretamente à pergunta / especificação :
 - Não responde facilmente: Qual a operadora atual do celular X ? (**requer busca**)
 - Não consegue facilmente garantir a unicidade telefone, operadora, data.



- Você foi por este caminho ?



- Você não necessariamente falhou porque:
 - Conseguiu responder às perguntas / especificações :
 - Qual a operadora do celular X na data YYYY/MM/DD ?
 - Qual a operadora do celular X na data YYYY/MM/DD ?
 - Consegue garantir a unicidade de 1 celular por operadora por data.

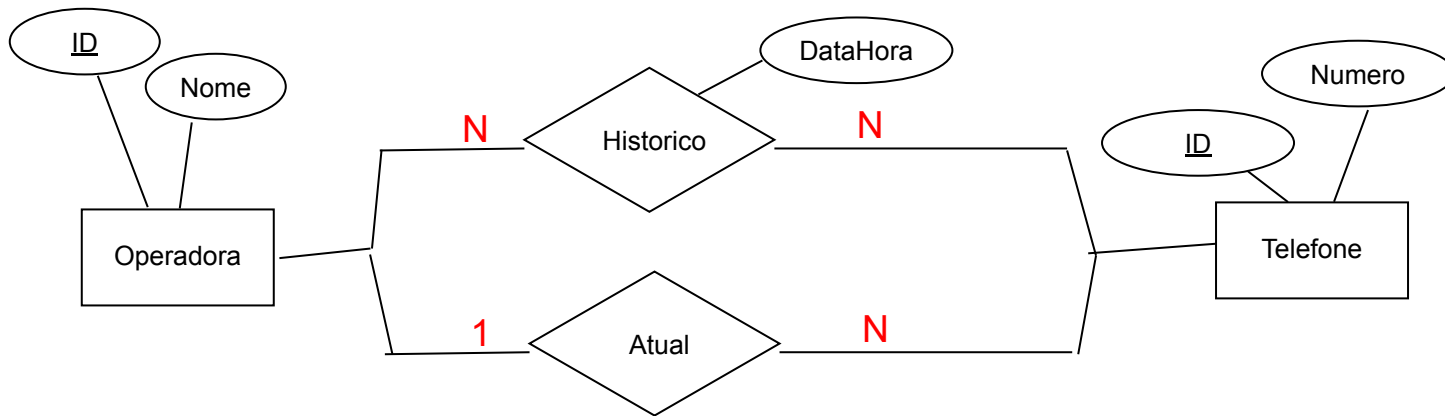
Mas.... seu modelo ainda ficou, digamos... Complexo.

Mas, vamos testá-lo no modelo lógico relacional.





- O que acha desta solução ?



Telefone
<u>Id_Telefone</u>
Numero
Id_operadora

Tem
<u>Id_Telefone</u>
<u>Id_Operadora</u>
Data

Operadora
<u>Id</u>
Nome

Operadora	
Id	Nome
1	VIVO
2	TIM

Tem / Portabilidade_Para		
Id_telefone	Id_Operadora	Data
1	1	15/04/2010
2	1	05/12/2012
1	2	10/10/2001
2	2	25/12/2016

Telefone		
Id_telefone	Numero	Id_Operadora
1	5511912345678	2
2	5511987654321	2
3	...	
4	...	



- Vamos testar este modelo ?

Telefone	Tem	Operadora
Id_Telefone Numero Id_operadora	Id_Telefone Id_Operadora Data	Id Nome

Em 5 mins:

- Relembre como funciona o MSSQL
 - Conexão no banco, execução de comandos, janelas e botões

Depois, Em 10 mins:

- Crie as estruturas acima no MSSQL (só crie as PKs)





- Você tipou o número do telefone assim ?

```
CREATE TABLE Telefone (  
    Id_telefone INT NOT NULL  
    , Numero INT NOT NULL  
    , Id_operadora INT NOT NULL  
)
```

Um Int vai de -2147483648 a 2147483647

Um telefone **5511912345678** só seria aceito em um BIG INT
que vai de -9223372036854775808 até 9223372036854775807

Ou, em um CHAR(13), mas, qual dos dois ?

Por que não VARCHAR nem NCHAR ou NVARCHAR ?

Se vocês erraram: **DROP TABLE** Telefone;





Seu modelo ficou parecido com este ?

```
CREATE TABLE Telefone (  
    Id_telefone INT NOT NULL IDENTITY(1,1)  
    , Numero CHAR(13) NOT NULL  
    , Id_operadora INT NOT NULL  
    , CONSTRAINT PK_telefone PRIMARY KEY (Id_telefone)  
)  
GO  
CREATE TABLE Operadora (  
    Id INT NOT NULL IDENTITY(1,1)  
    , Nome VARCHAR(255) NOT NULL  
    , CONSTRAINT PK_Operadora PRIMARY KEY (Id)  
)  
GO  
CREATE TABLE OperadoraTelefone (  
    Id_Telefone INT NOT NULL  
    , Id_Operadora INT NOT NULL  
    , Data Date NOT NULL  
    , CONSTRAINT PK_OperadoraTelefone  
        PRIMARY KEY (Id_Telefone, Id_Operadora)  
)
```





A primary key de operadortelefone está ok?

```
CREATE TABLE OperadoraTelefone (  
    Id_Telefone INT NOT NULL  
    , Id_Operadora INT NOT NULL  
    , Data Date NOT NULL  
    , CONSTRAINT PK_OperadoraTelefone  
        PRIMARY KEY (Id_Telefone, Id_Operadora)  
)
```

id	celular	operadora	data
1	912345678	vivo	10/10/2010
2	912345678	tim	05/12/2012
3	912345678	vivo	21/01/2014

Não ? Então, como arrumar ?





```
DROP TABLE OperadoraTelefone;
```

```
GO
```

```
CREATE TABLE OperadoraTelefone (  
    Id_Telefone INT NOT NULL  
    , Id_Operadora INT NOT NULL  
    , Data Date NOT NULL  
    , CONSTRAINT PK_OperadoraTelefone  
        PRIMARY KEY (Id_Telefone, Id_Operadora, Data)  
)
```

```
OU
```

```
CREATE TABLE OperadoraTelefone (  
    Id INT NOT NULL IDENTITY(1,1)  
    , Id_Telefone INT NOT NULL  
    , Id_Operadora INT NOT NULL  
    , Data Date NOT NULL  
    , CONSTRAINT PK_OperadoraTelefone  
        PRIMARY KEY (Id)  
)
```





Para conferir suas tabelas:

`SELECT * FROM operadora`

`SELECT * FROM telefone`

`SELECT * FROM OperadoraTelefone`

`SP_HELP operadora`

`SP_HELP telefone`

`SP_HELP OperadoraTelefone`

****OU****

Marque apenas o nome do tabela (como se se fosse executar parte do código SQL e digite ALT+F1



Exemplo:

Ex: `sp_help operadora`

Name	Owner	Type	Created_datetime
Operadora	dbo	user table	2019-04-15 18:48:40.550

Column_name	Type	Computed	Length	Prec	Scale	Nullable	TrimTrailingBlanks	FixedLenNullInSource	Collation
Id	int	no	4	10	0	no	(n/a)	(n/a)	NULL
Nome	varchar	no	255			no	no	no	Latin1_General_CI_AS

Identity	Seed	Increment	Not For Replication
Id	1	1	0

RowGuidCol
No rowguidcol column defined.

Data_located_on_filegroup
PRIMARY

index_name	index_description	index_keys
PK_Operadora	clustered, unique, primary key located on PRIMARY	Id

constraint_type	constraint_name	delete_action	update_action	status_enabled	status_for_replication	constraint_keys
PRIMARY KEY (clustered)	PK_Operadora	(n/a)	(n/a)	(n/a)	(n/a)	Id

Table is referenced by foreign key
fit_alunos.SALAS\1700666.Telefone: fkID_operadora



- Em 10 mins, utilizando ALTER TABLE vamos:
 - Adicionar as FKs (Foreigns Keys)





Ficou algo assim ?

```
ALTER TABLE OperadoraTelefone  
ADD CONSTRAINT FK_Telefone ( Id_Telefone )  
REFERENCES Telefone ( Id_telefone )
```

```
ALTER TABLE OperadoraTelefone  
ADD CONSTRAINT FK_Operadora ( Id_Operadora )  
REFERENCES Operadora ( Id_Operadora )
```

```
ALTER TABLE Telefone  
ADD CONSTRAINT FK_Operadora ( Id_Operadora )  
REFERENCES Operadora ( Id_Operadora )
```

Não tem algo de errado neste script ?





```
ALTER TABLE OperadoraTelefone  
ADD CONSTRAINT FK_Telefone FOREIGN KEY ( Id_Telefone )  
REFERENCES Telefone ( Id_telefone )
```

```
ALTER TABLE OperadoraTelefone  
ADD CONSTRAINT FK_OperadoraTelefone FOREIGN KEY ( Id_Operadora )  
REFERENCES Operadora ( Id )
```

```
ALTER TABLE Telefone  
ADD CONSTRAINT FK_Operadora FOREIGN KEY ( Id_Operadora )  
REFERENCES Operadora ( Id )
```

Nome da coluna em Operadora é Id.

Duas regras não podem ter o mesmo nome.

Syntaxe errada, faltou a palavra FOREIGN KEY





- Em 5 mins, utilizando ALTER TABLE vamos:
 - Adicionar a regra de validação do tipo de número de telefone.





Ficou algo assim?

```
ALTER TABLE telefone
  ADD CONSTRAINT CK_numeroTelefone
  CHECK ( numero
like '55[1-9][0-9][9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]' )
```

OU

```
ALTER TABLE telefone
  ADD CONSTRAINT CK_numeroTelefone
  CHECK ( numero between 5510900000000 and 5599999999999 )
```

?





- Em 5 mins, utilizando ALTER TABLE vamos:
 - Adicione a regra de unicidade para tentar garantir que um telefone não exista ao mesmo tempo em duas operadoras.





Ficou algo assim ?

```
ALTER TABLE OperadoraTelefone  
ADD CONSTRAINT UQ_RegraAnatel  
UNIQUE ( id_telefone, Data )
```

Será que só isso é o suficiente ?

O que mais eu precisaria para garantir isso ?

Dependendo da solução, ela não ficou igual à PK ?

Algumas regras não podem ficar no banco e precisam ser controladas externamente (aplicação)





Vamos testar o modelo:

- Em 10 mins, utilizando INSERTs vamos:
 - Adicionar alguns telefones (quaisquer 2 números)
 - Adicione as operadoras do Brasil:
 - Vivo.
 - Claro.
 - TIM.
 - Oi.
 - Nextel.
 - Algar.
 - Sercomtel.
 - MVNO's (Porto Seguro, Datora e Terapar)





Se você começou pelo telefone, rodando:

```
INSERT INTO Telefone ( numero )  
values ( '5511912345678' ), ( '5511987654321' )
```

Provavelmente teve este erro:

Msg 515, Level 16, State 2, Line 54

Cannot insert the value NULL into column 'Id_operadora', table 'fit_alunos.dbo.Telefone'; column does not allow nulls. INSERT fails.

The statement has been terminated.

Ou seja, temos que primeiro cadastrar as operadoras por causa da integridade referencial (PK -> FK) entre telefone e operadora.





Então, as operadoras ficaram assim:

```
Insert into operadora ( nome )
```

```
values
```

```
( 'Vivo' )
```

```
, ( 'Claro' )
```

```
, ( 'TIM' )
```

```
, ( 'Oi' )
```

```
, ( 'Nextel' )
```

```
, ( 'Algar' )
```

```
, ( 'Sercomtel' )
```

```
, ( 'MVNO' 's (Porto Seguro, Datora e Terapar)' )
```





Agora, testando os dados inseridos:

SELECT * **FROM** operadora

	Id	Nome
1	1	Vivo
2	2	Claro
3	3	TIM
4	4	Oi
5	5	Nextel
6	6	Algar
7	7	Sercomtel
8	8	MVNO's (Porto Seguro, Datora e Terapar)





Agora, de volta aos Telefones:

```
INSERT INTO Telefone ( numero, Id_operadora )  
values ( '5511912345678', 1 ), ( '5511987654321', 2 )
```

Supondo que o primeiro celular(id=1) seja da vivo(id=1) e o segundo(id=2) da claro(id=2).

Será que só isso é o suficiente ?





Melhorar também inserir no histórico :

```
insert into OperadoraTelefone  
(Id_Telefone, Id_Operadora, DATA )  
select 1, 1, convert(date, getdate() )
```

```
insert into OperadoraTelefone  
(Id_Telefone, Id_Operadora, DATA )  
select 2, 2, '20190422'
```

Cuidado que os Ids podem não bater.

Cada Insert com erro reserva e 'pula' o ID.

Ou seja, é normal a tabela apresentar 'buracos'





Agora, em 5 mins, realize a portabilidade do primeiro número da vivo (id=1) para Tim (id=3)

Registre esta portabilidade na data de hoje.





Primeiro arrumamos o histórico:

```
insert into OperadoraTelefone  
(Id_Telefone, Id_Operadora, DATA )  
select 1, 3, convert(date, getdate() )
```

Depois arrumo a operadora atual daquele numero:

```
update telefone set Id_Operadora = 3  
where Id_telefone = 1 and Id_operadora = 1
```





E conferimos os resultados:

```
select * from telefone
```

```
select * from OperadoraTelefone
```

	Id_telefone	Numero	Id_operadora
1	1	5511912345678	3
2	2	5511987654321	2

	Id_Telefone	Id_Operadora	DATA
1	1	1	2019-04-15
2	1	3	2019-04-15
3	2	2	2019-04-15

Lembra de nossa regra de unicidade ?

Veja como o numero de id=1 está em 2 operadoras (1 e 3) na mesma data (2019-04-15)...





Vamos corrigir as datas para evitar duplicidade:

Suponha que a portabilidade tenha sido agendada para 20/04/2019

```
UPDATE OperadoraTelefone set data = '2019-04-20'  
where Id_Operadora = 3 AND Id_Telefone = 1
```

E conferir :

```
select * from telefone
```

```
select * from OperadoraTelefone
```

	Id_telefone	Numero	Id_operadora
1	1	5511912345678	3
2	2	5511987654321	2

	Id_Telefone	Id_Operadora	DATA
1	1	1	2019-04-15
2	1	3	2019-04-20
3	2	2	2019-04-15





Agora vamos tentar resolver as questões iniciais:

- Qual a operadora atual do celular **5511912345678** ?

```
SELECT OPERADORA.NOME AS OPERADORA
FROM OPERADORA
INNER JOIN TELEFONE
ON OPERADORA.Id = TELEFONE.Id_operadora
WHERE TELEFONE.Numero = '5511912345678'
```

	OPERADORA
1	TIM





Agora vamos tentar resolver as questões iniciais:

- Qual a operadora do celular **5511912345678** na data **2019-04-15**?

```
SELECT OPERADORA.NOME AS OPERADORA
FROM OPERADORA
INNER JOIN OperadoraTelefone AS rel
on operadora.id = rel.Id_Operadora
INNER JOIN Telefone
on rel.Id_Telefone = telefone.Id_telefone
WHERE TELEFONE.Numero = '5511912345678'
and data <= '2019-04-15'
```

	OPERADORA
1	Vivo





Agora vamos tentar resolver as questões iniciais:

- Quais as portabilidades (de – para) realizadas pelo celular **5511912345678** desde **2019-04-15** ?

```
SELECT OPERADORA.NOME AS OPERADORA, rel.DATA
FROM OPERADORA
INNER JOIN OperadoraTelefone AS rel
on operadora.id = rel.Id_Operadora
INNER JOIN Telefone
on rel.Id_Telefone = telefone.Id_telefone
WHERE TELEFONE.Numero = '5511912345678'
and data >= '2019-04-15'
```

	OPERADORA	DATA
1	Vivo	2019-04-15
2	TIM	2019-04-20

Ele não trouxe uma lista de de-para, apenas as portabilidades.
Só é possível obter esta lista, com comandos “avançados” de SQL





Agora vamos tentar resolver as questões iniciais:

- Quais os celulares atuais da operadora Y ?

```
SELECT TELEFONE.Numero  
FROM OPERADORA  
      INNER JOIN TELEFONE  
      ON OPERADORA.Id = TELEFONE.Id_operadora  
WHERE OPERADORA.NOME = 'Vivo'
```

Numero





Agora que vocês aprenderam / revisão toda a parte de modelagem, criação e testes do modelo, podemos partir para novas experimentações, que incluem:

- Testar outros modelos
- Aprimorar o modelo escolhido adicionando funcionalidades.

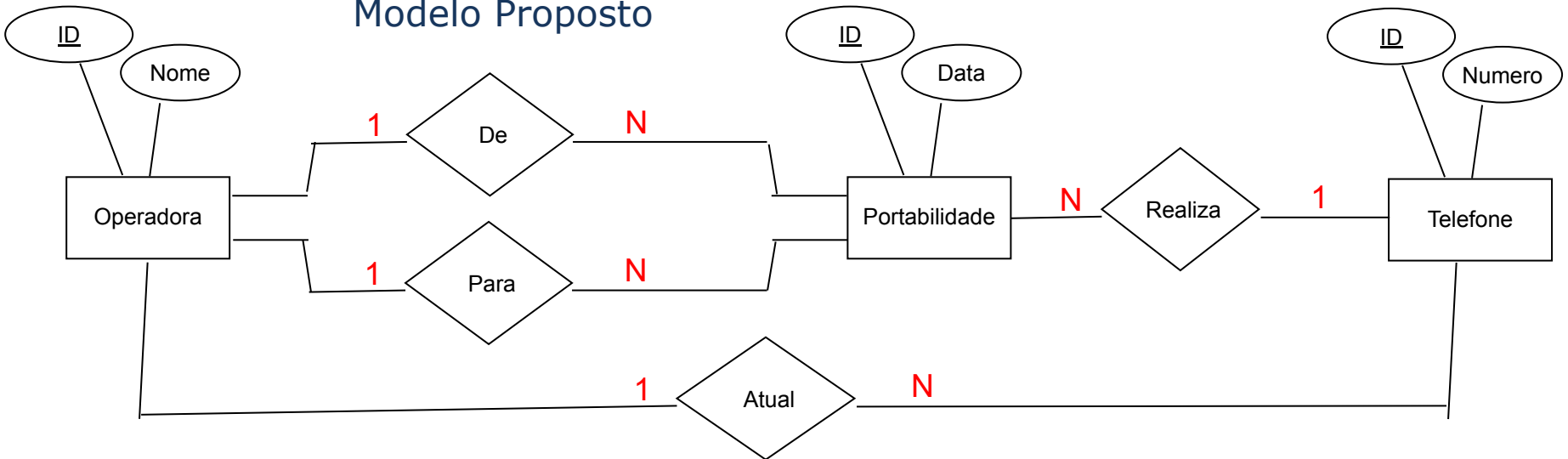
E se:

- Todo telefone pertencesse à Anatel antes de ser portado para sua primeira operadora
- Toda portabilidade registra-se a operadora anterior (de) e posterior (para).





Modelo Proposto



Telefone
<u>Id_Telefone</u>
Numero
Id_operadora

Portabilidade
<u>Id_Telefone</u>
<u>Id_Operadora_De</u>
<u>Id_Operadora_Para</u>
Data

Operadora
<u>Id</u>
Nome

Operadora	
Id	Nome
0	Anatel
1	VIVO
2	TIM

Portabilidade			
Id_telefone	Id_Operadora_De	Id_Operadora_Para	Data
1	0	1	10/10/2010
2	0	1	05/12/2012
3	1	2	15/04/2001
4	1	2	25/12/2016

Telefone		
Id_operadora	Numero	Id_Operadora
1	5511912345678	2
2	5511987654321	2
3	...	
4	...	



Crie, teste e veja se este último modelo atende à especificação.

Imagine outras formas de modelar este mesmo problema, teste-as, se encontrar alguma mais eficiente, compartilhe com a turma.





Faculdade
IMPACTA
TECNOLOGIA

Obrigado!

Aula Gravada por:

Prof. Msc. Gustavo Bianchi Maia

gustavo.maia@faculdadeimpacta.com.br

