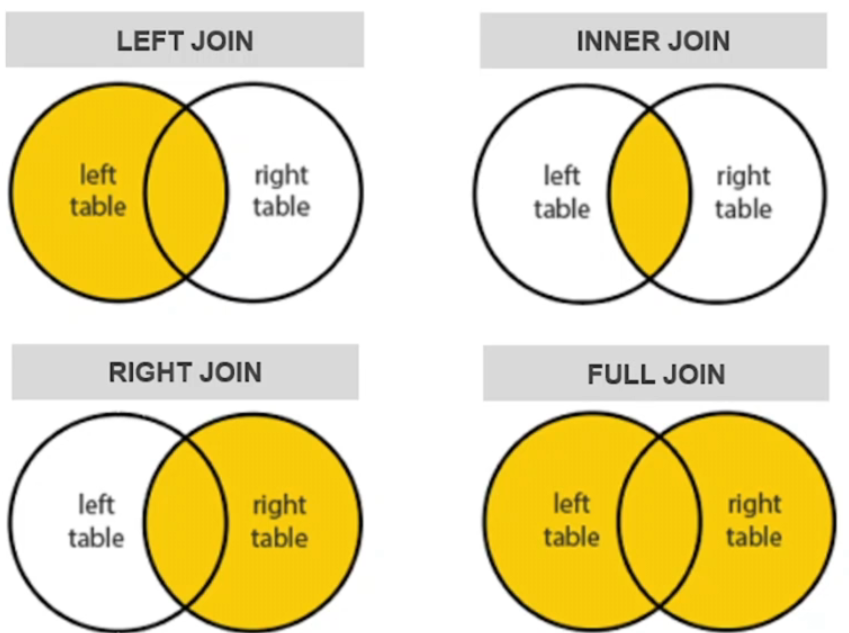
TIpos de JOINS



Exemplo 1: Utilize o LEFT JOIN para fazer join entre as tabelas

– temp\_tables.tabela\_1 e temp\_tables.tabela\_2

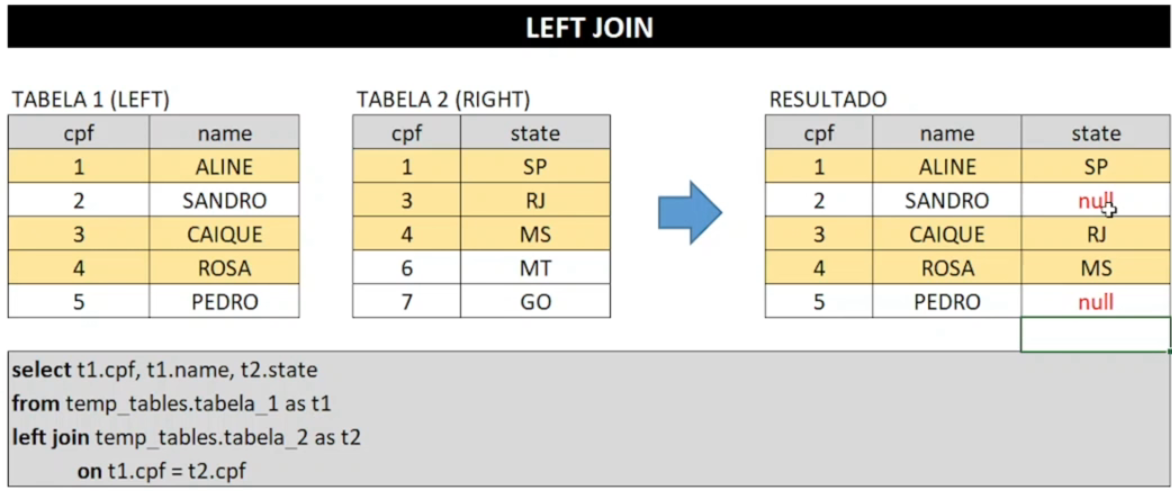
select \* from temp\_tables.tabela\_1

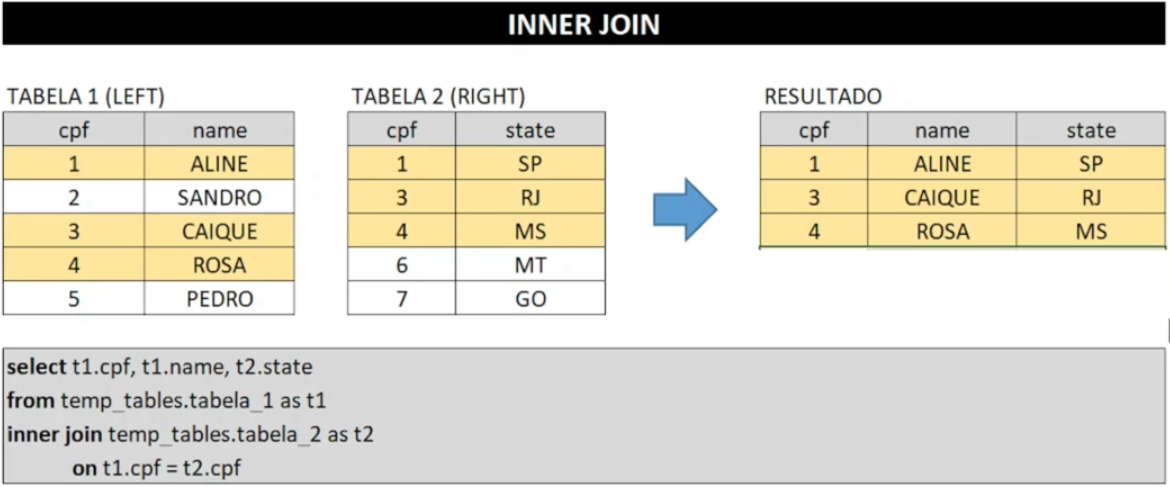
select \* from temp\_tables.tabela\_2

select t1.cpf, t1.name, t2.state

from temp\_tables.tabela\_1 as t1 left join temp\_tables.tabela\_2 as t2

on t1.cpf = t2.cpf





Exemplo 3: Utilize o RIGHT JOIN para fazer join entre as tabelas

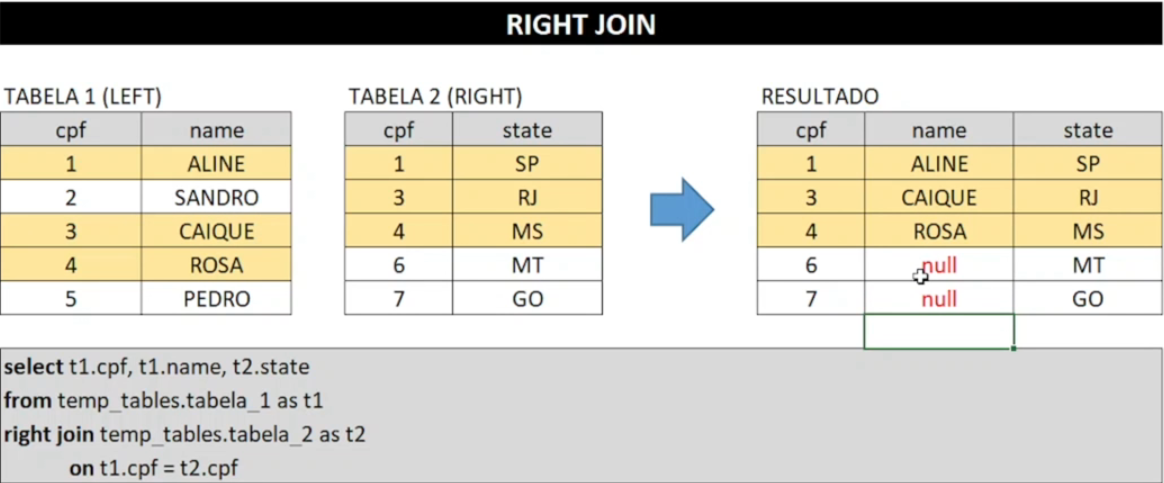
– temp\_tables.tabela\_1 e temp\_tables.tabela\_2

select t1.cpf, t1.name, t2.state

from temp\_tables.tabela\_1 as t1

right join temp\_tables.tabela\_2 as t2

on t1.cpf = t2.cpf



Exemplo 4: Utilize o FULL JOIN para fazer join entre as tabelas

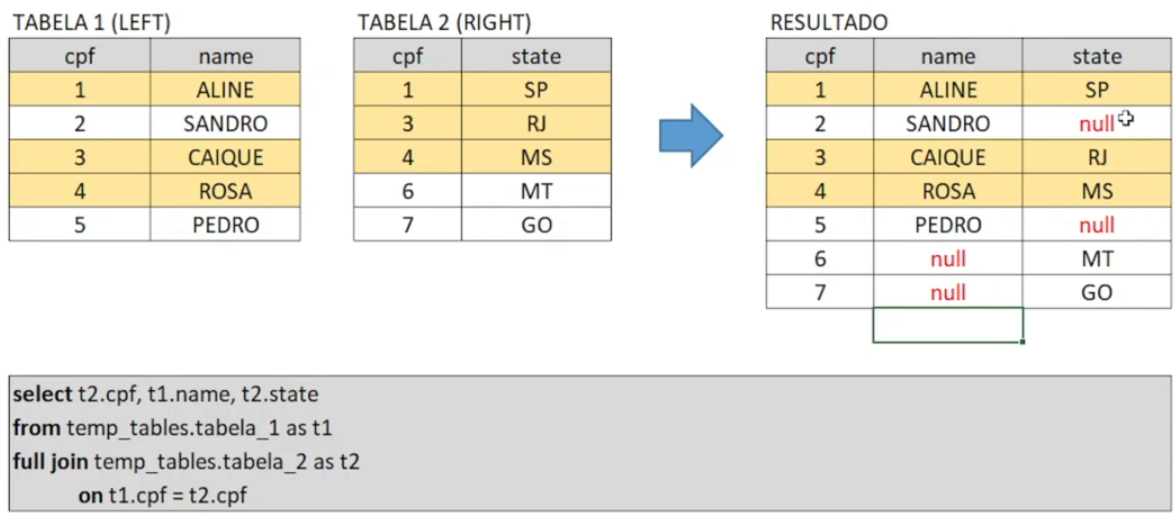
– temp\_tables.tabela\_1 e temp\_tables.tabela\_2

select t2.cpf, t1.name, t2.state

from temp\_tables.tabela\_1 as t1

full join temp\_tables.tabela\_2 as t2

on t1.cpf = t2.cpf



Exemplos :

1. Identifique qual é o status profissional mais frequente nos clientes que compraram automóveis em nosso site

select cus.professional\_status,

count (fun.paid\_date) as pagamentos

from sales.funnel as fun

left join sales.customers as cus

on fun.customer\_id = cus.customer\_id

group by cus.professional\_status

order by pagamentos desc

1. Identifique qual é o gênero mais frequente nos clientes que compraram automóveis no site. OBG: utilizar a tabela temp\_tabela.ibge\_genders

select \* from temp\_tabela.ibge\_genders limit 10

select ibge-gender, count(fun.paid\_date)

from sales.funnel as fun

left join sales.customers as cus

on fun.customer\_id = cus.customer\_id

left join temp\_tables.igbe\_genders as ibge

on lower(cus.first\_name) = ibge.first\_name

group by ibge.gender

1. Identifique de quais regiões são os clientes que mais visitam o site

* utilizar a tabela temp\_tables.regions

select \* from sales.customers limit 10

select \* from temp\_tables.regions limit 10

from sales.funnel as fun

left join sales.customers as cus

on fun.customer\_id = cus.customer\_id

left join temp\_tables.regions as reg

on lower(cus.city) = lower(reg.city)

and lower(cus.state) = lower(reg.state)

group by reg.region

order by visitar desc