5.5 - Fazendo consultas em SQL

00:00:00:00 - 00:00:00:57
Olá, pessoal. Na aula passada, aprendemos consultas básicas. Nessa aula, vamos ver alguns comandos mais complexos para complementar nossas consultas: o join o group by e o order by. Bom, passamos por comandos e consultas importantes que uma pessoa analista de dados precisa saber. Vamos agora aprender a juntar duas tabelas. Temos um comando para fazer isso que se chama join. O comando join é usado para combinar dados de duas ou mais tabelas em um banco de dados. Um join no SQL é similar ao Merge que aprendemos lá no Pandas para juntar dois arquivos. Assim como no Merge, precisamos definir o nosso "how", o tipo de junção que faremos entre os dados. E nós temos diferentes tipos. Nessa imagem nós conseguimos visualizar os tipos de join e eu vou explicar um por um. Mas já começamos supondo que temos duas tabelas, cada uma representada por um círculo. e a gente quer fazer o join dessas tabelas, ou seja, queremos consultar os dados das duas tabelas fazendo uma combinação.

00:00:00:57 - 00:00:01:58
Então, primeiro a gente tem o inner join, esse primeiro aqui, que vai retornar apenas esse ponto em vermelho, que é o encontro das duas tabelas, ou seja, que são os registros que têm correspondência nas duas tabelas. A gente tem o full join, que está aqui do lado direito, que, como podem ver na imagem, retorna todos os registros das duas tabelas. A gente tem o left join, que está aqui no inferior esquerdo. Esse comando retorna todos os registros da tabela à esquerda e os registros que têm correspondência na tabela direita. Vejam aqui na imagem: toda a tabela da esquerda está de vermelho, inclusive o ponto de encontro entre as duas tabelas. Então, tem de retorno a tabela esquerda inteira e esse ponto de correspondência aqui entre as duas tabelas. E a gente tem o right join, o right de direita em inglês. O right join é parecido com o left join, só que agora obtemos como retorno a tabela direita. A gente pode ver aqui na imagem: a gente tem a tabela direita toda em vermelho e inclusive o ponto de encontro entre as duas tabelas também.

00:00:01:58 - 00:00:02:56
É muito importante a gente entender essas diferenças entre os tipos de joins, pois independente de qual join usarmos, nós vamos obter resultados, mas precisamos saber se esse é realmente o resultado que estamos buscando. Vamos voltar lá para o nosso banco de dados, aqui nas nossas três tabelas e vamos combinar os dados das tabelas municípios brasileiros e município status. Vamos supor que queremos a população residente em cada cidade. Assim, nós podemos combinar as tabelas municípios brasileiros e município status utilizando aquela coluna em comum que é o município ID. Então, a gente começa aí com a nossa query. Vou comentar essa última query aqui para começar logo abaixo. A gente começa com select, que a gente quer fazer uma consulta, selecionar tudo, né? Nós queremos a coluna cidade, que está na tabela municípios brasileiros. Então, a gente coloca o nome da tabela um ponto e o nome da coluna. Assim, ó: municípios brasileiros ponto e qual que é a coluna que a gente quer?

00:00:02:56 - 00:00:04:19
Cidade. Ok. E queremos também a coluna de população residente que está na tabela de município status. Então, a gente coloca essa vírgula e coloca o nome da tabela, ponto, e o nome da coluna. Então, município status, ponto população residente. Ok. Agora nós usamos o from para dizer qual tabela que estamos consultando. Então, eu vou colocar aqui: from. Estamos inicialmente consultando a municípios brasileiros. Então, a gente vai colocar: from municípios brasileiros. Municípios brasileiros. Vou dar um Enter aqui para continuar para ficar melhor de a gente visualizar. Ah, vocês viram que foi criado, assim que eu coloquei "from municípios brasileiros", já foi criado aqui um MB. Esse MB é como se fosse um apelido para essa tabela. E aí, toda vez que a gente fosse usar municípios brasileiros, a gente colocava MB no lugar do nome da tabela. Só para poder ficar melhor de a gente visualizar e saber de qual tabela que a gente está falando aqui, eu vou apagar e vou continuar chamando municípios brasileiros, beleza?

00:00:04:19 - 00:00:05:37
Então, vou dar Enter para poder ficar mais fácil aqui de continuar. E agora a gente vai usar o inner join para retornar apenas os registros onde há correspondência entre os IDs de município das duas tabelas. Porque nós temos a coluna de município ID nas duas tabelas, então a gente pode usar essa coluna de referência para juntar os dados. Então, a gente coloca ali o comando do inner join. inner join. E informamos qual tabela que estamos realizando esse join. Então, a gente coloca from municípios brasileiros inner join município status. Aí, de novo, ele já apareceu aqui como MS para poder dar um apelido para esta tabela. Como eu vou preferir escrever o nome da tabela completo, eu só apago e pronto. A gente está especificando a condição de junção usando a cláusula on. Então, inner join tabela on, mas qual que é a coluna, né? É isso que a gente está querendo falar. On. Aqui a gente está dizendo que a gente quer combinar os registros do valor da coluna cidade na tabela municípios brasileiros, onde é igual os valores das colunas de município ID das duas tabelas. Então, a gente coloca: on municípios brasileiros, a gente coloca o ponto município ID, vou já colocar aqui, igual a município status ponto município ID.

00:00:05:37 - 00:00:07:01
Isso nos permite obter informações sobre a população residente de cada cidade, combinando os dados das duas tabelas com base no ID de município. Então, só recapitulando tudo aqui, a gente quer selecionar da tabela de municípios brasileiros a coluna de idade e da de município status a população residente. Porque vocês lembram que na tabela de municípios status a gente não tem o nome da cidade? A gente só tem o nome da cidade lá na tabela de municípios brasileiros? Então beleza, a gente quer a cidade de uma tabela e a população residente de outra. Como que a gente faz isso? A gente faz isso fazendo o inner join. O inner join a gente está colocando aqui: inner join da tabela de municípios status onde o município ID de uma seja igual ao município ID de outra, ok? Então, a gente pode executar essa tabela e ver o resultado. E aqui está. Essa query retorna apenas o registro onde há correspondência entre os IDs de município nas duas tabelas. E como temos o nome da cidade em uma tabela e a população residente em outra, nós conseguimos combinar essas informações das duas tabelas e obter esse resultado aqui.

00:00:07:01 - 00:00:08:02
Onde podemos observar que, por exemplo, na cidade de Acrelândia tem 2.538 habitantes, ok? Que tal vocês tentarem fazer o join para retornar os nomes dos municípios que filtramos na aula passada? Ah, pessoal, uma coisa é que lá no Python, a gente estava falando que era sempre o ponto, função. O ponto sempre meio que antecedia uma função. Aqui vocês podem perceber que quando a gente usa ponto, é mais uma referência. Então, a gente está, por exemplo, falando que a tabela é tal, ponto, coluna X. Então, esse ponto aqui já é diferente daquele que a gente usava lá no Python. Beleza? A gente viu aqui o exemplo do inner join. No nosso caso, não teria diferença se a gente usasse o right join, o left join ou o full join, porque as nossas colunas de IDs nas duas tabelas são iguais. Ou seja, todos os IDs têm correspondência. Mas quando formos juntar com os dados que já temos, sabemos que não temos pessoas de todos os municípios do país. Então, saber a diferença entre os joins ou merges no Pandas fará diferença para nós.

00:00:08:02 - 00:00:09:05
Vamos deixar materiais complementares sobre esse assunto aqui para vocês. Vamos fazer uma outra consulta agora. Vamos supor que a gente queira fazer uma contagem do número de cidade para cada estado na tabela Municípios Brasileiros. Então nós queremos de resultado o nome do estado e a quantidade de cidade que tem nesse estado, né? Então, vamos lá. Vou comentar essas duas linhas aqui e vou dar um Enter para começar a minha consulta embaixo. A gente começa com o Select. A gente quer o estado, né? Então, Select Estado vírgula. Ok, queremos o estado, mas queremos a quantidade de cidades também. Para isso, nós usamos uma função de agregação que conta o número de ocorrências da coluna cidade. Essa função se chama count, de contagem mesmo. E dentro dos parênteses, a gente vai colocar aqui qual é a coluna que a gente quer fazer essa conta. Então, a gente coloca count cidade, que a gente quer fazer a soma da cidade por estado. Então, estamos selecionando a coluna estado e fazendo a soma das cidades.

00:00:09:05 - 00:00:10:24
Beleza. Aí, depois disso, a gente coloca from, e a gente informa qual que é a tabela que a gente está fazendo toda essa pesquisa aí. From municípios brasileiros. Ok. E, por fim, nós vamos agrupar o resultado usando o comando group by. Vamos agrupar por qual coluna? Pela coluna de estado, não é? Então, aqui estamos informando que queremos contar o número de ocorrências de cada cidade para cada valor único da coluna de estado. Então, por exemplo, temos o estado de Minas Gerais. Cada cidade cujo estado seja Minas Gerais, ou MG, vai ser contabilizado pelo count. Então, finalizando nossa query, a gente tem from, e aí a gente coloca o group by Estado. Executando essa consulta, a gente tem aqui esse retorno. Então, conseguimos ver na primeira coluna o estado e na segunda a contagem de cidades por estado. Mas suponha que queremos saber qual estado tem mais cidades. Seria legal se a gente ordenasse essa quantidade aqui, né? Nós temos um comando que é o order by, usado para ordenar os resultados da consulta. Precisamos indicar o que queremos ordenar, no caso queremos ordenar o resultado da contagem de cidades.

00:00:10:24 - 00:00:11:40
Primeiro nós temos o resultado de estados e depois a contagem de cidade por estado. Então vamos indicar que queremos ordenar o segundo item da nossa consulta, correto? Que a gente quer ordenar pela maior quantidade, né? O primeiro são os estados e o segundo é a contagem. E para finalizar, a gente vai indicar se queremos ordenar em ordem crescente ou decrescente. Vamos ordenar de forma decrescente. Assim, vamos visualizar no topo da tabela qual o estado com mais cidade. Bom, então, voltando aqui para a nossa query, a gente tem um group by estado, depois a gente coloca o order by, A gente coloca dois, porque a gente está querendo ordenar o segundo item aqui do nosso select, que é a cidade. E a gente coloca o dez, opa, dez, de decrescente, ok? E vamos executar. E agora a gente consegue ver aqui, visualizar de forma mais fácil, que o meu país, Minas Gerais, é o estado que tem mais cidades. Logo em seguida vem SP e por aí vai, ok? Bom, vamos consultar agora a tabela de gerência região. Suponha que queremos saber a quantidade total de pessoas brancas e pretas, independente da região. Para isso, nós usamos o select de novo, mas dessa vez usamos o comando sum, que basicamente faz a somatória para nós. Então, vamos lá.

00:00:11:40 - 00:00:13:17
Vou comentar isso daqui. Vou no final, dar um enter, select sum. Queremos as somas de brancos e pretos, né? Então, vamos ter dois sums. Bom, aqui dentro dos parênteses, a gente vai colocar o nome da coluna. Então, a gente pode vir aqui no canto esquerdo e abrir aqui gerência região, para a gente poder ver o nome da coluna, para a gente poder colocar o nome certinho, né? Tem que ser certinho o nome. Então, o nome da coluna é pessoas brancas. Então, vamos colocar: pessoas brancas. E o segundo sum que a gente tá querendo saber é de gerência branca. Ah, na verdade é de pessoas pretas e pardas. A gente quer saber a soma total das pessoas, né? Tô me confundindo. Pessoas pretas e pardas. Ok. Tá certinho, tá certinho. Opa, tá faltando, tem um S aqui a mais. Agora sim. E agora colocamos o from para dizer de qual tabela a gente está querendo esse resultado: from gerência região. Aqui, como eu apertei ali pra poder já completar direto, de novo ele aparece aqui com esse apelidinho pra tabela. Como a gente vai sempre escrever o nome da tabela completa, a gente pode apagar, ok? Então, beleza. Então, a gente tem a soma de pessoas brancas, a soma de pessoas pretas e pardas da tabela de gerência região.

00:00:13:17 - 00:00:14:52
Vamos executar pra gente poder ver o resultado. E aqui ao executar a query, a gente tem esse resultado. Uma coluna informando que a soma de pessoas brancas é 43.349 e uma coluna informando que a soma de pessoas pretas e pardas é 52.580. Agora, vamos encontrar qual a região com o maior número de pessoas pretas e pardas. Então, vamos usar o Select de novo, né? Vou comentar essa daqui e aqui no final apertar Enter Select. Mas dessa vez a gente quer de retorno a região e o maior número de pessoas pretas no total, né? Então, a gente pode usar o comando de max, que é o max, ok? Que retorna o maior valor. Então, a gente vai pegar primeiro a região, a gente quer primeiro a região, depois, sim, o max de pessoas pretas pardas, ok? From... Opa, fora do parênteses: from gerência região. Recapitulando, a gente está querendo qual que é a região que tem o máximo de pessoas pretas e pardas na tabela de gerência região, ok? Então executando, a gente consegue aqui de resultado que é a região sudeste que tem a maior quantidade de pessoas pretas e pardas. Beleza? Podemos fazer o contrário também.

00:00:14:52 - 00:00:16:13
Encontrar a região com o menor número de pessoas pretas e pardas. No lugar do max, a gente coloca o min. Assim: no lugar do max aqui, a gente coloca o min. Executa e pronto. A gente vê que o resultado é a região sul, indicando que a região sul apresenta a menor quantidade de pessoas pretas e pardas. E podemos, por fim, consultar quais regiões o número de brancos na gerência é maior que o número de pretos na gerência, pretos e pardos na gerência. Então a gente vai lá. Vamos comentar aqui. A gente começa com o quê? Adivinhem. Select. E aí a gente quer região, né? Então select região from gerência região. Aqui. E queremos todas as regiões em que a quantidade de brancos na gerência seja maior que pretos e pardos na gerência, né? Ou seja, é uma condição. Então, a gente usa o where. E aqui, para fazer essa comparação de maiores, a gente usa aquele símbolo matemático. Então, a gente coloca where gerência branca maior que gerência preta e parda. Como apareceu aqui a opção para mim, eu já posso clicar aqui para poder completar. Ok? E aí a gente pode executar para ver o resultado.

00:00:16:13 - 00:00:16:34
Então, recebemos de resultado as regiões em que a quantidade de brancos na gerência é maior do que de pretos e pardos na gerência. A gente tem como as regiões sudeste, sul e centro-oeste. Legal, né? Aprendemos comandos importantes em SQL para fazer consultas. Na próxima aula, vamos juntar isso com os dados que temos para deixar a nossa análise mais completa. Até lá.

progra{m}aria