

06

## Criar variáveis com condicionais

### Transcrição

[00:00] Construímos nossa base de cadastro de clientes com informações de fidelidade, mas não fizemos nenhum tipo de discriminação. Temos a data de retirada e de retorno, o custo de reparo, se o cliente causar danos no jogo. A intenção é fazer uma premiação com os clientes, mas se ele quebrar ou atrasar várias entregas, não vale a pena. Precisamos de uma forma de analisar se tudo deu certo.

[01:47] Todos os contratos são feitos com período de trinta dias. Se o cliente demorou mais de trinta dias para entregar o jogo, significa que ele atrasou. Teríamos que desconsiderar esse cliente na informação. E se teve algum custo de reparo, também teríamos que desconsiderar. Podemos inclusive usar essa informação para melhorar nosso relacionamento com o cliente. Por exemplo, um alugou quinze jogos, mas só cinco não apresentam problemas. Poderíamos mandar um e-mail avisando para ele a data de entrega, por exemplo.

[03:15] Vamos começar criando um código auxiliar antes de voltarmos para o código final. Precisamos criar por exemplo uma variável que diga se a operação é válida para promoção ou não. Antes fazíamos a variável de marcação usando um condicional, mas vimos como fazer condicionais no data step, não no SQL. Dá para fazer.

[04:05] Vamos começar pedindo para criar uma tabela com contratos válidos por CPF. Depois, selecionamos as variáveis com asterisco. Antes, vamos criar uma variável para nos auxiliar, que é de intervalo de tempo em que o cliente ficou com os jogos. Fazemos isso com a data de retorno menos a data de retirada. Essa variável vai ser chamada dias. Também vamos criar a variável condicional. O condicional no SQL é mais parecido com o select statement do que com o if then else. No data step usávamos muito o if then else, mas aqui é outra forma. Falamos que caso seja isso, faça outra operação, com a expressão case e when, colocando a expressão toda depois do when. Quando dias for maior que 30, que significa que teve o atraso, precisamos de outra condição. Ou se o custo de reparo for maior que zero. Então, then, operação inválida, que é zero. Caso contrário, é uma operação válida, chamamos de um. Aqui também temos que lembrar de fechar colocando o end. Vamos usar o from para pegar da base de operações.

[08:36] Ele diz que a variável dias não foi encontrada na base. É bem parecido, porque a variável dias realmente não está na base. Nós que construímos agora. Isso quer dizer que não podemos usar no condicional? Podemos. Só precisamos avisar para o SQL que a variável não está na base. Ela foi calculada no nosso processo. Falamos isso com calculated antes do nome dela.

[09:45] Ainda precisamos dar o nome para a variável, senão o SAS inventa um aleatório. Para isso, usamos o as depois do END. Como é uma variável de marcação, vamos chamar de flag\_valida.

[10:35] Nossa variável agora diz se a operação é válida para promoção. Conseguimos construir nossa marcação condicional.