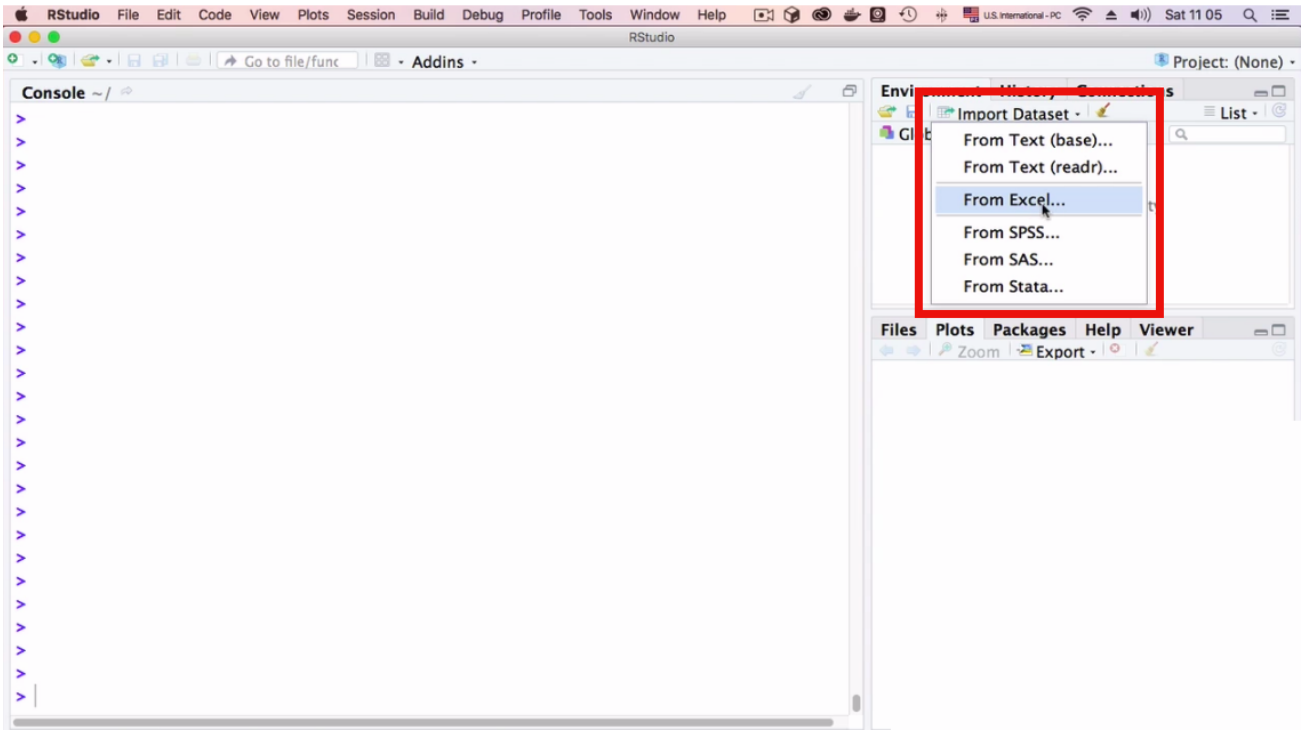
04

Importação e exibição dos dados

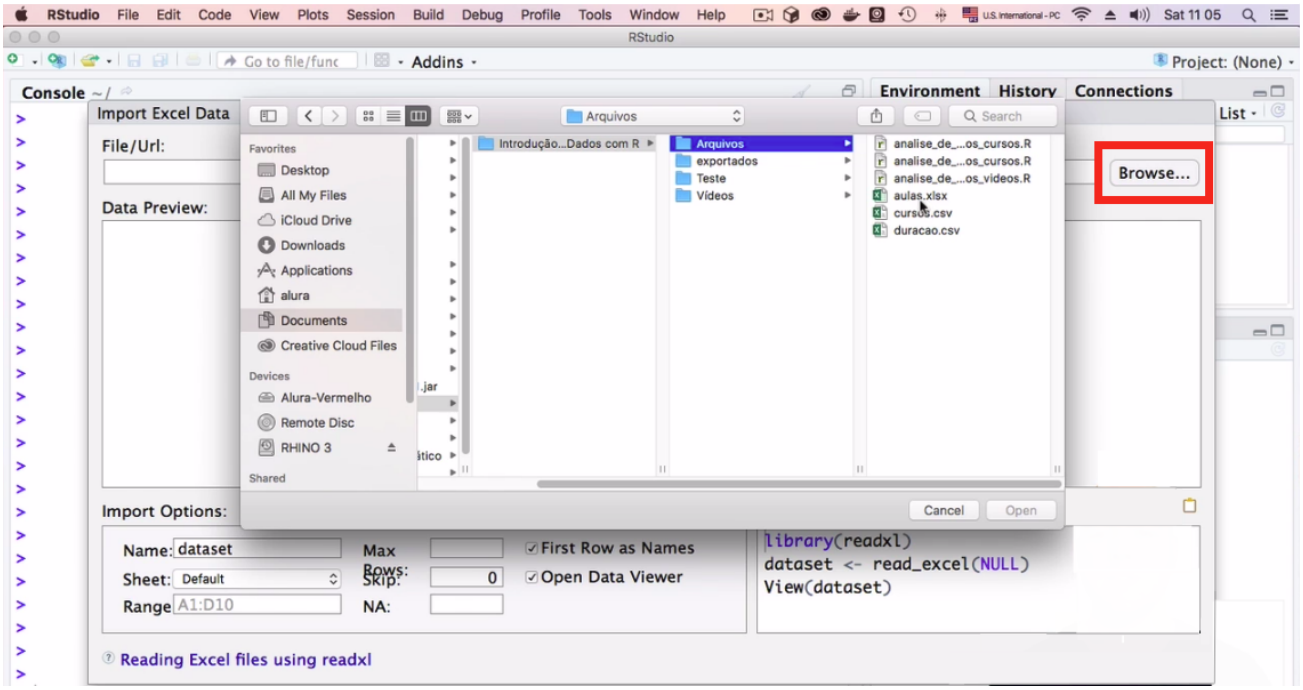
Transcrição

No projeto hipotético que trabalharemos, somos contratados por uma empresa de cursos online, para analisar quantitativamente seus vídeos. Assim, forneceremos informações a ela, como a popularidade, quanto tempo os alunos levam para concluir os cursos, entre outros.

O primeiro passo será importar o banco de dados, disponível em Excel, para o RStudio. Para isso, clicaremos no botão "*Import Dataset*", localizado na janela superior à direita, e selecionaremos a opção *From Excel*.

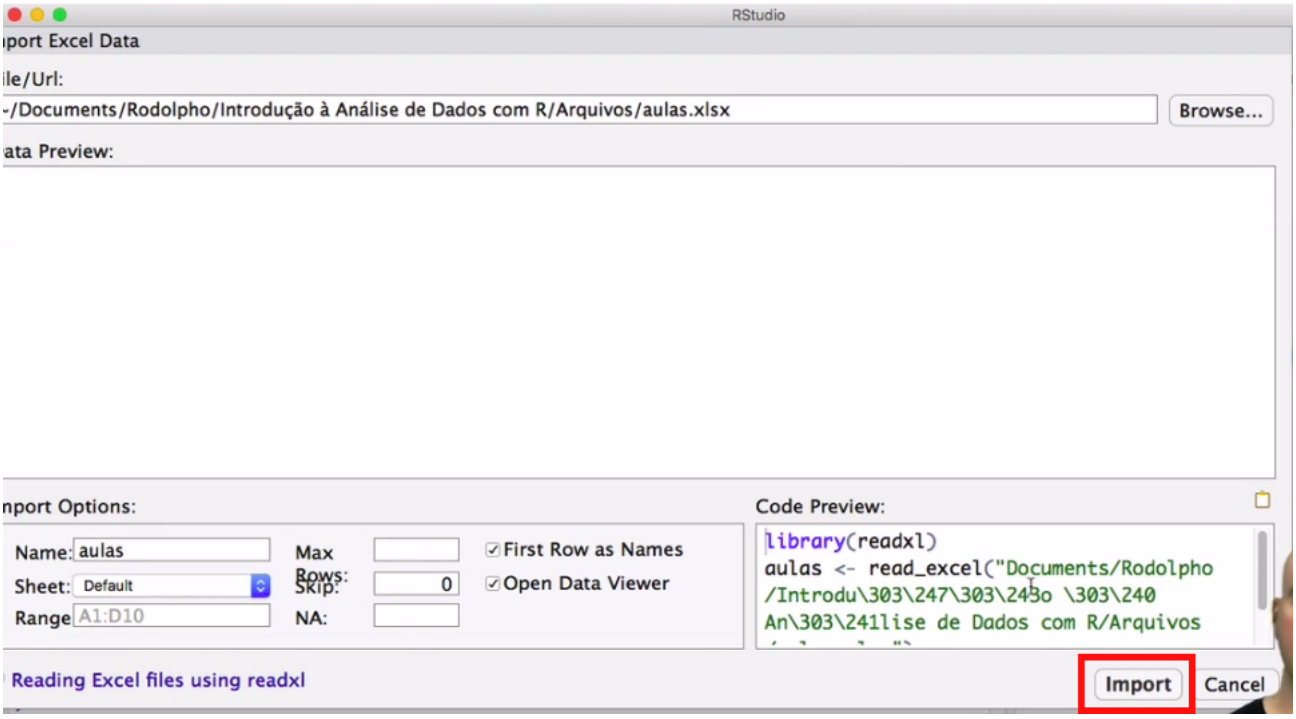


Feito isso, uma janela se abrirá, na qual poderemos buscar documentos, clicando em *Browse*. Procuraremos o banco de dados `aulas.xlsx` , disponibilizado para download (https://cursos.alura.com.br/course/business-analytics-com-r/task/35171).

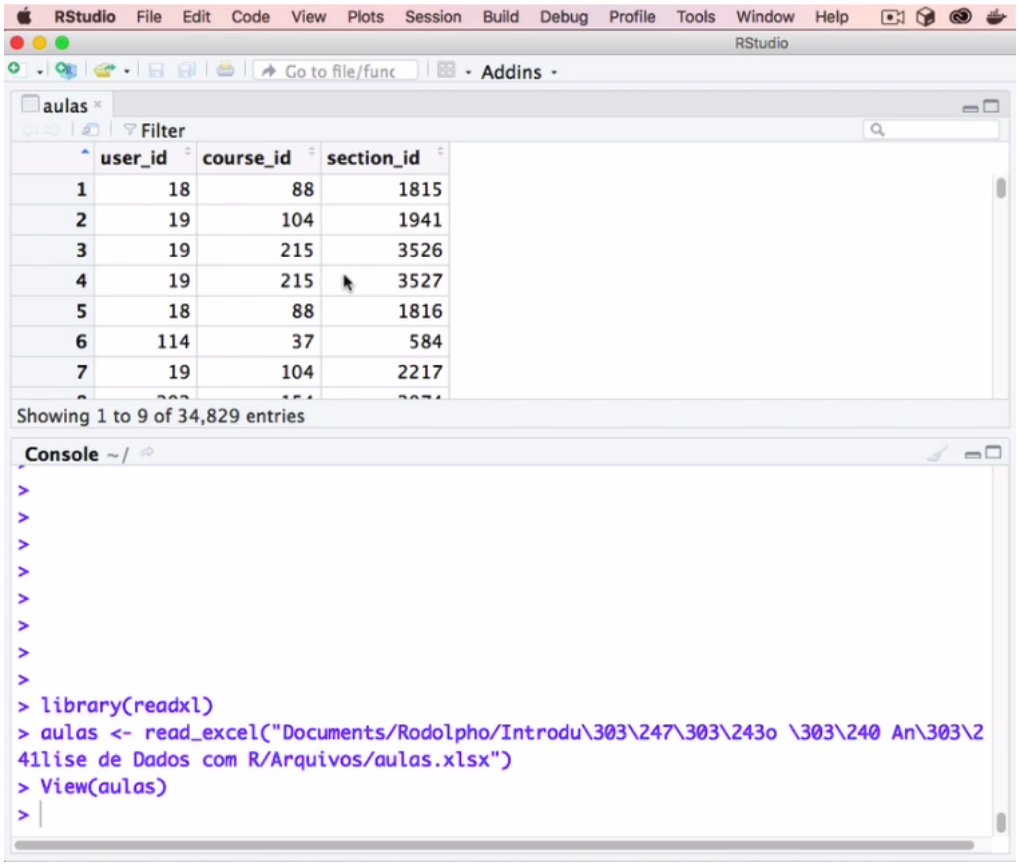


Selecionado o arquivo, caso apareça uma mensagem de erro, como a exibida abaixo, basta clicarmos em "OK". Isto verifica se o arquivo é mesmo em Excel.

Em seguida, clicaremos no botão "*Import*":



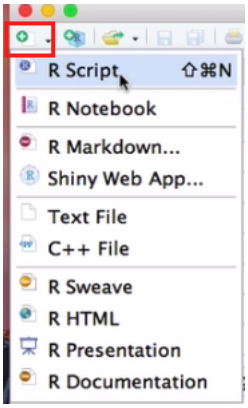
Na tela do programa, serão exibidas as variáveis e observações. Lembrando que a edição de planilhas é melhor no Excel. No RStudio é possível editá-las, mas ele é mais adequado para trabalho estatístico.



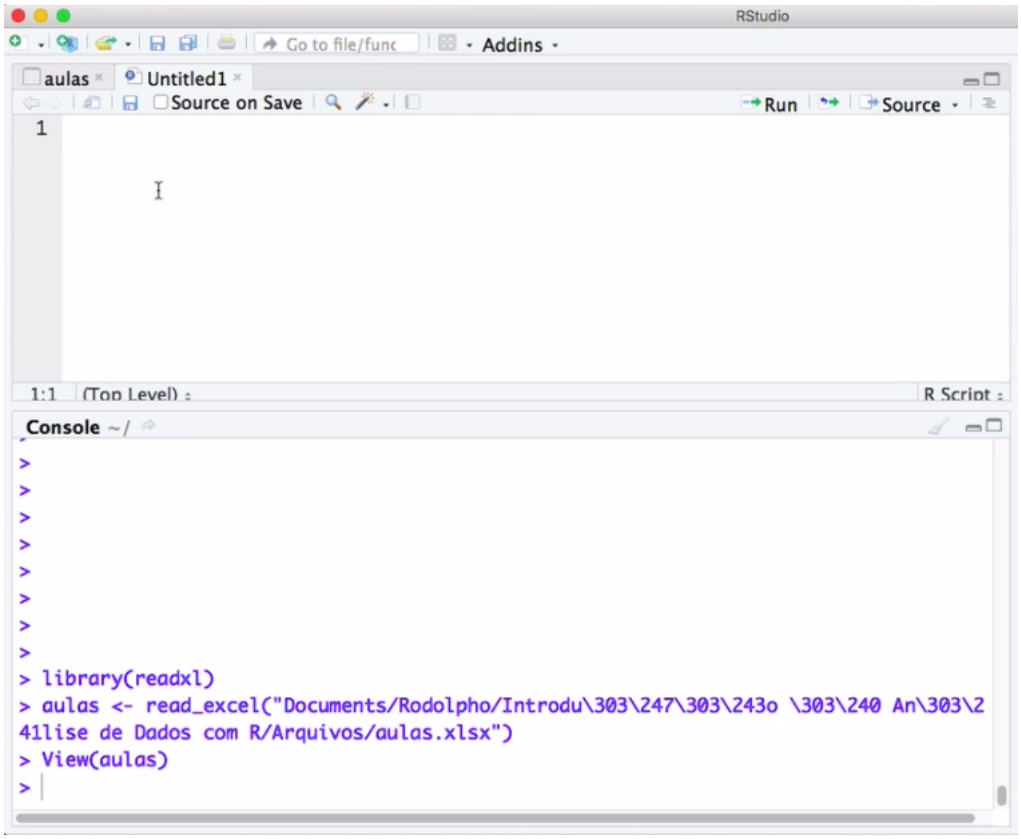
Importado o banco de dados, teremos na parte superior da tela a planilha com as 3 variáveis da amostra:

- `user._id` , código atribuído ao aluno;
- `course_id` , código do curso;
- `section_id` , código para os vídeos.

Por exemplo, na primeira linha da tabela, a informação é que o usuário `18` fez o curso `88` e assistiu o vídeo `1815` . Começaremos a trabalhar clicando no ícone verde com sinal de mais (`+`) branco, localizado no canto superior esquerdo do programa, para criar um novo documento de "*R Script*".

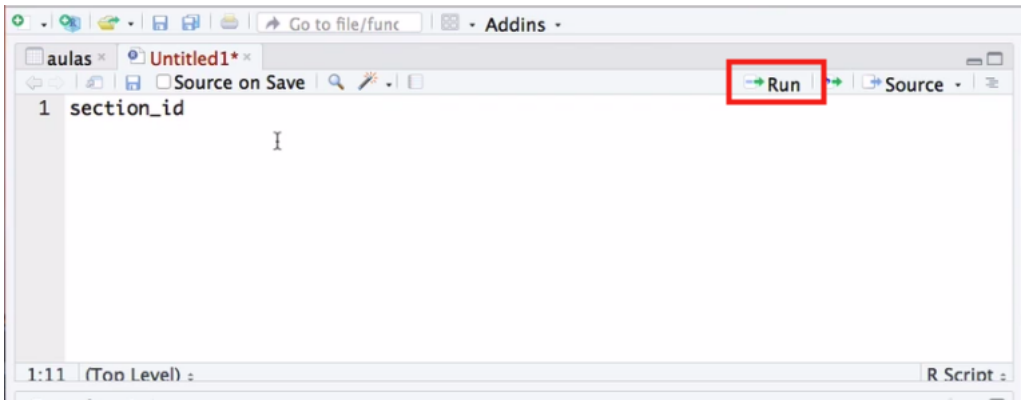


Será aberta uma nova aba com espaço em branco para digitarmos e editarmos comandos, sem executá-los automaticamente, como acontece em editores de texto.



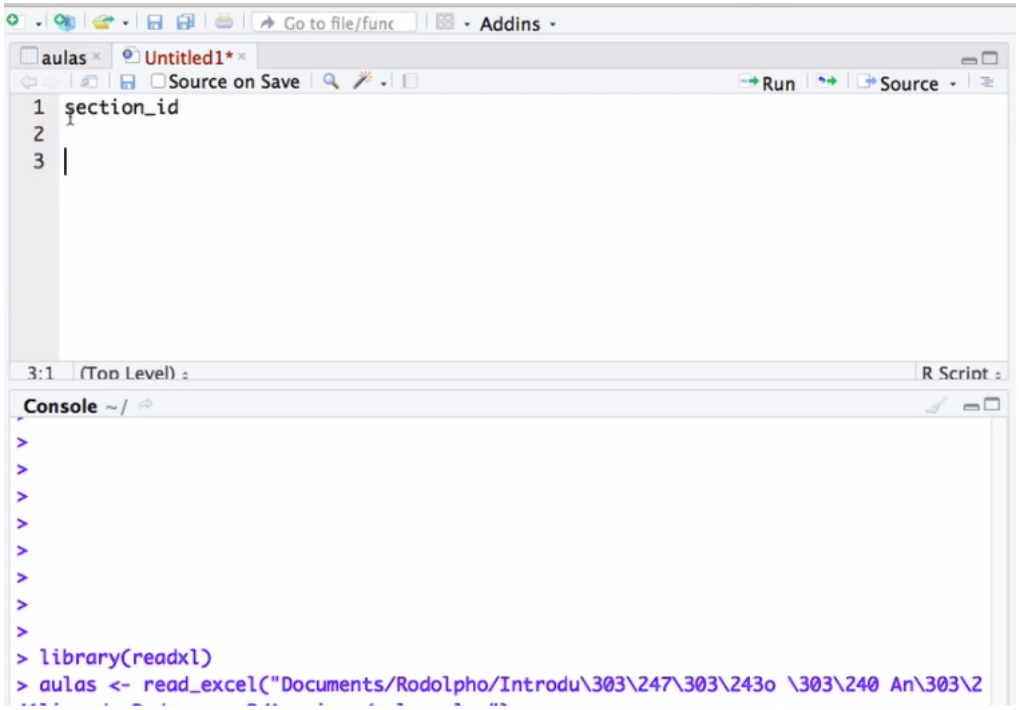
Os comandos que acrescentarmos nesse novo arquivo serão executados logo abaixo, no Console. Tentaremos isolar a variável `section_id`, para analisar quantas vezes cada vídeo foi assistido. Para isso, digitaremos no novo arquivo "R Script": `section_id`.

Para executar a informação, ou seja, para que R busque a variável, clicaremos no botão "*Run*" ou utilizamos o atalho "Command + Enter" (MAC) ou "Ctrl + Enter" (Windows).



Atenção: devemos prestar atenção ao local em que o cursor está antes de executarmos um comando.

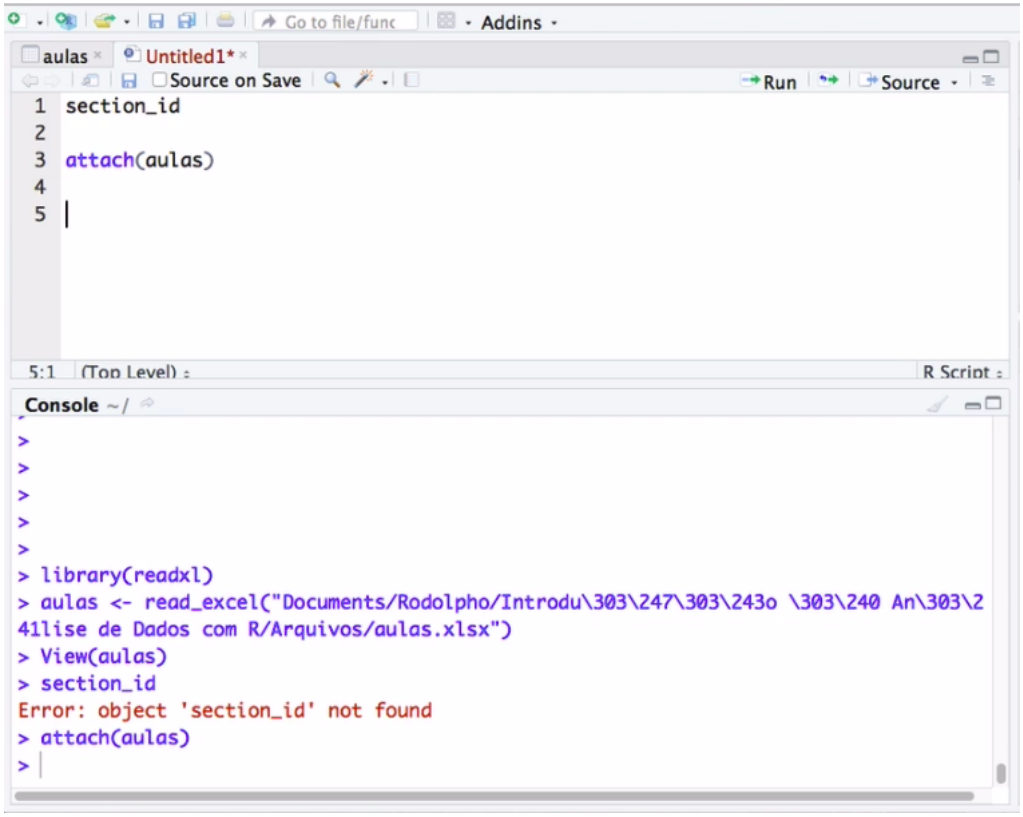
Se por exemplo digitarmos `section_id` na linha 1 e o cursor estiver na linha 3, que no momento está em branco, e clicarmos em *Run*, nada será executado.



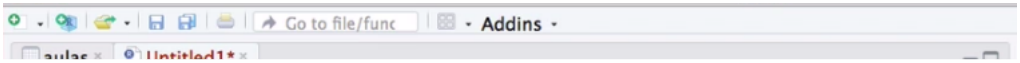
O cursor deve estar posicionado na linha a ser executada. Sendo assim, tentaremos executar `section_id`. O retorno que teremos no Console é de erro: `Error: object 'section_id' not found`.

Isso aconteceu porque R possui o banco de dados mas não reconhece o nome das variáveis contidas nele. Para que ele as reconheça, utilizaremos a função de anexação `attach()`, abaixo de `section_id` e, entre parênteses, especificaremos o banco de dados "aulas" `attach(aulas)`.

Notem que ao digitarmos `attach()` no Script, aparece a opção de autocompletar o comando que, ao executarmos, dessa vez utilizando o atalho "Command + Enter", nos trará o retorno correto no Console: `attach(aulas)`.



Então, tentaremos executar novamente a variável `section_id` .



No Console, serão exibidas algumas observações em relação à variável. Agora que o programa as reconhece, podemos trabalhar com o nome delas. Observem que na última linha do retorno no Console, há uma mensagem dizendo que foi atingido o número máximo de exibição:

```
[ reached getOption("max.print") -- omitted 33829 entries ]
```

O banco de dados `aulas.xlsl` contém muitas informações, apesar de ser uma amostra. O aviso aponta que foram omitidas `33829` entradas. Ou seja, estamos visualizando apenas um trecho de `section_id` . Para acessá-la por inteiro, teremos que alterar a opção `("max.print")` , o `default` de R, que estabelece o número de linhas que serão exibidas.

Para isso, utilizaremos a variável `options` com o parâmetro `max.print` , especificado com o número de linhas a serem exibidas entre parênteses. No Console, à direita, há um índice com a contagem de `991` linhas. Se `33829` estão omitidas, precisamos de, pelo menos, `34000` linhas. Sendo assim, estabeleceremos `40000` , no Sprint, para acessarmos todos os dados.

```
options(max.print = 400000)
```

Ao executarmos o comando, o padrão de exibição será alterado no Console. Se posicionarmos o cursor em `section_id` e clicarmos em "Run", todos os dados serão exibidos, e conseguiremos acessar o banco de dados. No entanto, as informações estão desordenadas, e isso dificulta a visualização dos tipos de dados.

Pode haver dados faltando ou fora do padrão, os quais não conseguiremos identificar no meio de tantos números. Quando isso acontece, os valores fora do padrão ficam no começo ou no final da coluna. Se quisermos visualizar apenas os primeiros valores da variável, podemos isolá-los utilizando a função `head` , que em inglês significa "cabeça", especificando-se `section_id` entre parênteses.

```
head(section_id)
```

Após executarmos o comando, teremos no Console:

```
> head(section_id)
[1] 1815 1941 3526 3527 1816 584
```

Foram exibidas as primeiras observações de `section_id` , aparentemente sem problemas. Mas a apresentação dos códigos continua desordenada, o que pode trazer erros imperceptíveis. Poderemos ordenar a exibição dos valores; por exemplo, se quisermos ordená-los do menor para o maior, utilizaremos a função `sort` no Script:

```
sort(section_id)
```

Ao executarmos o comando no Console, teremos a variável `section_id` ordenada pelo código dos vídeos, em ordem crescente.

Na última linha, encontraremos o valor `999999`, que destoa da amostra. Pode ser um valor incorreto, que entrou por erro ou que o desenvolvedor inseriu por engano, e ficamos sem saber do que se trata. Quando chegamos em um dado desse tipo, recorremos à empresa que solicitou a análise e informamos que há um dado fora do padrão, para que eles confiram e vejam o que houve. Caso seja um erro, solicitamos o valor correto da observação.