



Aula 04

Criando os primeiros scripts

Prof. Jalerson Lima

Apresentação

Nessa aula iremos aprender o conceito de *scripts*, além de construir os nossos primeiros *scripts* usando a linguagem Ruby. Por fim, vamos aprender como executar nossos *scripts* usando o Windows Explorer e o terminal de comandos.

Objetivos

1. Compreender o conceito de *script*;
2. Aprender como usar o Atom para criar *scripts* em Ruby;
3. Aprender como executar *scripts* usando o Windows Explorer e o terminal de comandos.

1. Introdução

Antes de começarmos a discutir a nossa aula, uma pergunta que você pode estar se fazendo é: o que é um *script*? Se você digitar a palavra “script” no Google (GOOGLE, 2015), ele dará a seguinte definição.

“Script é um texto com uma série de instruções escritas para serem seguidas, ou por pessoas em peças teatrais ou programas televisivos, ou executadas por um programa de computador. O termo é uma redução da palavra inglesa manuscript, que significa “manuscrito”, “escrito à mão”. (SIGNIFICADOS, 2015).

Um *script* é um arquivo de texto que contém instruções escritas em alguma linguagem de programação. Essas instruções podem ser lidas por algum interpretador, que irá executá-las e produzir algum resultado ou efeito. No nosso caso, nós vamos criar *scripts* usando a linguagem Ruby, e iremos utilizar o interpretador Ruby (ruby.exe).

2. Criando os primeiros scripts em Ruby

Para criar um nosso *script*, podemos utilizar um editor de textos qualquer como o Bloco de Notas do Windows, contudo recomendamos o uso do Atom, a ferramenta de desenvolvimento que ensinamos a instalar na nossa primeira aula. Mas fique à vontade para usar uma ferramenta de sua preferência.

Crie um novo arquivo usando o Atom ou seu editor de código preferido. No Atom, clique no menu *File* (canto superior esquerdo) e depois em *New File*. O Atom irá abrir uma nova janela permitindo que você possa editar o conteúdo desse novo arquivo. Digite o código abaixo conforme ilustra o Exemplo de código 1.

1	<code>puts "Olá mundo!"</code>
---	--------------------------------

Exemplo de código 1 - Olá Mundo!

Observe que no Exemplo de código 1 incluímos a numeração das linhas do código, portanto você não precisa digitar o número que aparece no início de cada linha. Após digitar o código acima, salve o arquivo clicando em *File* (canto superior esquerdo) e depois em *Save*. Uma janela será aberta para que você indique o nome e o local onde o arquivo deverá ser salvo. Salve o arquivo com o nome “exemplo1.rb” (sem aspas) no local onde você preferir. Você pode usar outro nome para o seu arquivo, contudo ele precisa ter a extensão “.rb”, pois é isso o que vai indicar para o nosso Windows que esse arquivo é um *script* escrito em Ruby.

Conforme ilustra a Figura 1, após salvar o seu *script*, algumas mudanças serão evidentes:

- O código ficou com uma coloração. Chamamos isso de *syntax highlight*, ou simplesmente de coloração de código;
- O nome do *script* apareceu na aba. No caso da Figura 1 o nome do *script* é “exemplo1.rb” (sem aspas);

- O local onde nosso *script* foi salvo aparece no canto inferior da tela.

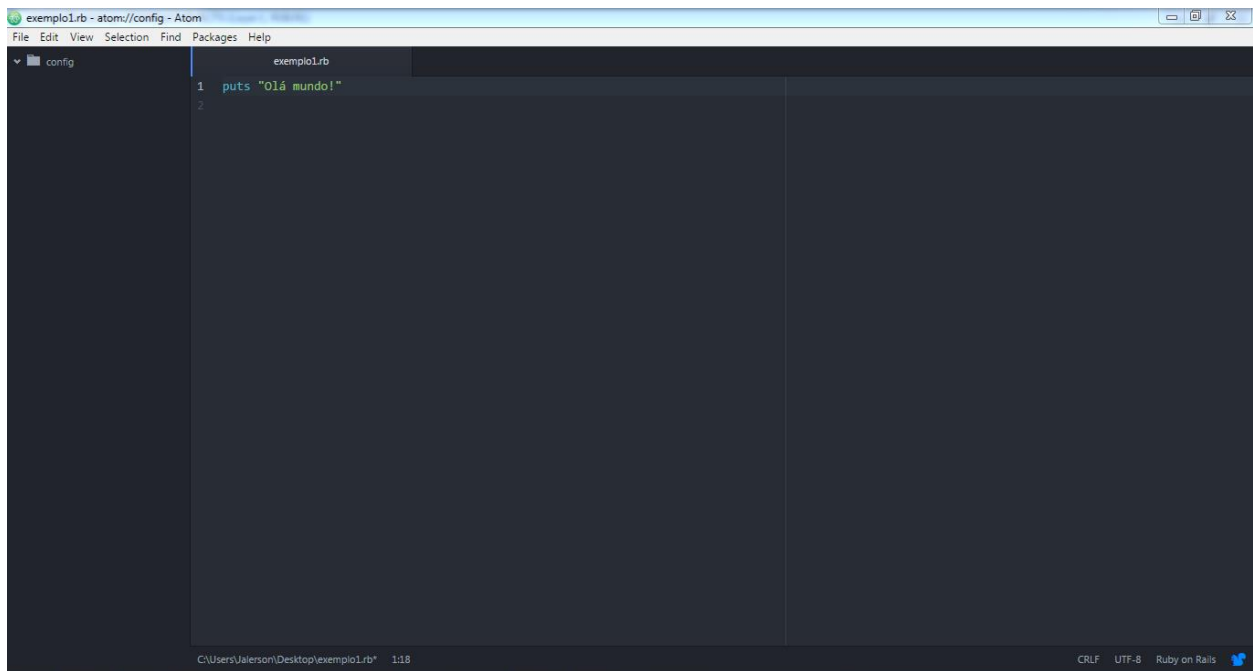


Figura 1 - Olá Mundo no Atom

3. Executando scripts

Existem duas formas de executar *scripts*: a primeira delas é através do Windows Explorer (o gerenciador de arquivos do Windows) e a segunda é usando um terminal de comandos.

3.1 Executando scripts com o Windows Explorer

Para executar seu *script*, primeiro abra a pasta onde você salvou o seu *script* usando o Windows Explorer (o gerenciador de arquivos do Windows). O seu arquivo deve ser exibido conforme ilustra a Figura 2.

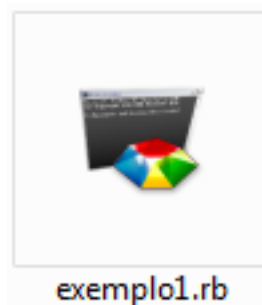


Figura 2 - *Script* no Windows Explorer

Após localizar o seu *script* no Windows Explorer, basta clicar duas vezes sobre ele para executá-lo.

Atividade 4.1

Crie um *script* Ruby usando o Atom ou seu editor de código preferido e salve-o com o nome “aula04-atividade01.rb” (sem aspas). Digite o código apresentado no Exemplo de código 1, salve o *script* novamente e execute-o.

Você vai perceber que uma tela preta será exibida rapidamente e depois irá fechar sozinha. Isso ocorre porque o *script* executa muito rapidamente, porque há uma única instrução para ser executada. Ao concluir a execução do *script*, o Windows simplesmente fecha a janela, impedindo que possamos ver o resultado. Para resolver esse problema, vamos incluir mais uma linha de código no nosso *script*, conforme ilustra o Exemplo de código 2.

```
1 puts “Olá mundo!”  
2 gets
```

Exemplo de código 2 - Olá mundo!

Lembra a utilidade do método `gets`? Ele permite que o usuário digite algo no teclado para ser usado no nosso programa. Ao executar o método `gets`, o *script* fica aguardando que o usuário digite algo, vamos usar essa pausa na execução ao nosso favor, permitindo que observemos o resultado do nosso *script* antes que o Windows feche a janela. Após inserir a segunda linha de código, execute novamente o *script*.

Observe que dessa vez a janela ficou aberta. Para fechá-la, basta teclar ENTER.

Atividade 4.2

Altere o código do *script* que você criou na Atividade 4.1 incluindo a segunda linha de código, conforme apresenta o Exemplo de código 2. Salve o arquivo com o nome “aula04-atividade02.rb” e execute-o.

3.2 Executando scripts com o terminal de comandos

A segunda forma para executar nossos *scripts*, mais usada pelos programadores e por usuários do Linux, é usar o terminal de comandos do nosso sistema operacional. Se você conhecer os comandos usados no terminal, você pode optar por executar os *scripts* dessa forma. Caso não conheça, recomendamos que execute seus *scripts* através do Windows Explorer. A vantagem de executar os *scripts* usando o terminal de comandos é que essa forma dispensa o uso do `gets` no final do *script*, porque o Windows não irá fechar a janela do terminal ao finalizar a execução do *script*.

Para executar o *script* “exemplo1.rb” usando o terminal de comandos do Windows, primeiramente devemos abrir o programa “cmd.exe” (sem aspas), que é o próprio terminal de

comandos, e em seguida, devemos mudar para o diretório onde se encontra o nosso *script*, e por fim executamos o *script* usando o comando apresentado abaixo.

```
> ruby exemplo1.rb
```

No comando acima, estamos usando o interpretador do Ruby para executar o *script* “exemplo.rb”.

Resumindo

Nessa aula aprendemos o conceito de *script* e também como construir nossos próprios *scripts* usando a linguagem Ruby. Também aprendemos como executar nossos *scripts* usando o Windows Explorer e o terminal de comandos. Caso tenha restado alguma dúvida, não deixe de entrar em contato conosco através do Ambiente Virtual de Aprendizado.

Referências

GOOGLE. Google. **Google**, 2015. Disponível em: <<http://www.google.com>>. Acesso em: 02 nov. 2015.

SIGNIFICADOS. Script. **Significados**, 2015. Disponível em: <<http://www.significados.com.br/script/>>. Acesso em: 02 nov. 2015.

SOUZA, L. **Ruby - Aprenda a programar na linguagem mais divertida**. 1ª. ed. São Paulo: Casa do Código, v. I, 2012.