



Step Parameters e Step Data

Autor: Reinaldo M. R. Junior

Step Parameters

- ▶ No step você pode passar parâmetro para o step, usando as aspas duplas com o valor dentro, exemplo “reiload@gmail.com”.

Scenario: Cenário Negativo.

Given A pagina de login.

When Digitar o login.

And Digitar o password errado "1234".

And Efetuar a autenticacao.

Then Deve ser exibido uma mensagem de erro.

Step Data

- ▶ No step você pode passar informações/texto usando o keyword de comentário com três aspas duplas `"""` após o step, como no exemplo abaixo:

Scenario: Cenario Negativo.

Given A pagina de login.

When Digitar o login.

And Digitar o password errado `"1234"`.

And Efetuar a autenticacao.

Then Deve ser exibido uma mensagem de erro.

`"""`

`Try again! Wrong login name or password!`

`"""`

Exercício

- ▶ Apartir do exemplo anterior criar um cenário negativo passando o login por parâmetro.

Predefined Data Types in parse

- ▶ O behave tem um módulo de parse próprio, para ser feito de forma simples e adaptável. Que é inverso ao `string.format` do python, aonde você pode pré-definir o tipo de dado pra ele converter, o padrão do parse parâmetro é do tipo `string`.

Exemplo:

```
from behave import given
@given('I have {number:d} friends') #< Converte Número para o tipo int.
def step_given_i_have_number_friends(context, number):
    assert number > 0
```

Tabela com os tipos de dados suportado pelo parse!

Type	Characters Matched	Output Type
w	Letters and underscore	str
W	Non-letter and underscore	str
s	Whitespace	str
S	Non-whitespace	str
d	Digits (effectively integer numbers)	int
D	Non-digit	str
n	Numbers with thousands separators (, or .)	int
%	Percentage (converted to value/100.0)	float
f	Fixed-point numbers	float
e	Floating-point numbers with exponent e.g. 1.1e-10, NAN (all case insensitive)	float
g	General number format (either d, f or e)	float
b	Binary numbers	int
o	Octal numbers	int
x	Hexadecimal numbers (lower and upper case)	int
ti	ISO 8601 format date/time e.g. 1972-01-20T10:21:36Z	datetime
te	RFC2822 e-mail format date/time e.g. Mon, 20 Jan 1972 10:21:36 +1000	datetime
tg	Global (day/month) format date/time e.g. 20/1/1972 10:21:36 AM +1:00	datetime
ta	US (month/day) format date/time e.g. 1/20/1972 10:21:36 PM +10:30	datetime
tc	ctime() format date/time e.g. Sun Sep 16 01:03:52 1973	datetime
th	HTTP log format date/time e.g. 21/Nov/2011:00:07:11 +0000	datetime
tt	Time e.g. 10:21:36 PM -5:30	time



PEP8 + Behave + Python 3

PEP8 – Foco na Legibilidade do Código;

Behave – Foco na Descrição dos Testes de forma que todos possam entender;

Python 3 – Foco na Rapidez da criação dos Testes.

Referências:

- ▶ [1] <https://media.readthedocs.org/pdf/behave/latest/behave.pdf>

