

Programação Aplicada Ao Direito



U F *m* G



MARCELO CANDIOTTO
Sociedade de Advogados **MCSA**



ASSOCIAÇÃO
BRASILEIRA DE
**LAWTECHS &
LEGALTECHS**



Conteúdo

Dia 08 - Funções em Docassemble

- O que são funções
- Funções em Variáveis
- Funções de Data
- Funções de Texto
- Outros tipos de função
- Exercício Prático: Usando funções no seu projeto

Funções em Docassemble

Função

É um conjunto de comandos que realiza uma tarefa específica.

- Toda função tem **nome** e **()**
- Pode ou não ter argumentos e um valor de retorno
- Os parâmetros podem ser obrigatórios ou não

Ex.:

- `enviar_email("remetente@b.com", "destinatario@b.com", "Assunto", "Mensagem")`
- `data_atual()`
- `se(valor>0;pagar)`

Bloco de código

O Docassemble permite o uso de funções em Python, a imposição de pacotes externos e o controle da ordem das entrevistas.

```
28 ---  
29 code: |  
30     from num2words import num2words  
31     charge_value_in_words = num2words(charge_value, lang='pt_BR', to='currency')
```

Funções em Docassemble

Existem diversas funções prontas com diferentes aplicações no Docassemble. Você também pode criar as suas próprias funções e incluí-las em sua entrevista usando a linguagem Python e um bloco de módulos.

Para usar uma função em um módulo Python você deve importá-la. Existem 2 maneiras de importar o módulo inteiro:

```
---  
modules:  
  - datetime  
---
```

```
from datetime import *
```

Para importar só uma função do módulo, faça:

```
from datetime import datetime
```

Bloco de código python

Alterar a ordem da entrevista:

code: |

- é usado para guardar itens de uma lista.
- Exemplo: você pode criar a lista de UF's uma única vez e a chama na entrevista todas vez que for necessário.

```
metadata:  
  title: Alterando ordem da entrevista  
---  
mandatory: True  
question: |  
  Ordem da entrevista  
fields:  
  - "Defina o bloco:": author_name  
    datatype: radio  
    choices:  
      - A: A  
      - B: B  
---  
mandatory: True  
code: |  
  if author_name == 'A':  
    caminho_a  
  else:  
    caminho_b  
---  
question: |  
  Bloco A  
fields:  
  - "Nome A:": caminho_a  
---  
question: |  
  Bloco B
```

Funções em Docassemble

Exemplo de função em Docassemble

A palavra **green** é alterada para **Green** quando é usada a função *capitalize()*.

```
question: |  
    The word ${ color } becomes  
    ${ capitalize(color) } when  
    passed through the  
    `capitalize()` function.  
mandatory: True
```

Funções em variáveis

Função defined()

Objetivo: verifica se uma variável foi definida

Parâmetro: uma variável

Retorno: True ou False

```
question: Summary
subquestion: |
    Your favorite fruit is
    ${ favorite_fruit }.

    % if defined('favorite_vegetable'):
    Your favorite vegetable
    is ${ favorite_vegetable }.
    % else:
    I do not know your favorite
    vegetable.
    % endif
mandatory: True
```

Funções em variáveis

Função value()

Objetivo: mostrar o conteúdo da variável

Parâmetro: uma variável

Retorno: conteúdo da variável

```
question: Summary
subquestion: |
    Your favorite fruit is
    ${ value('favorite_fruit') }.
mandatory: True
```

Funções em variáveis

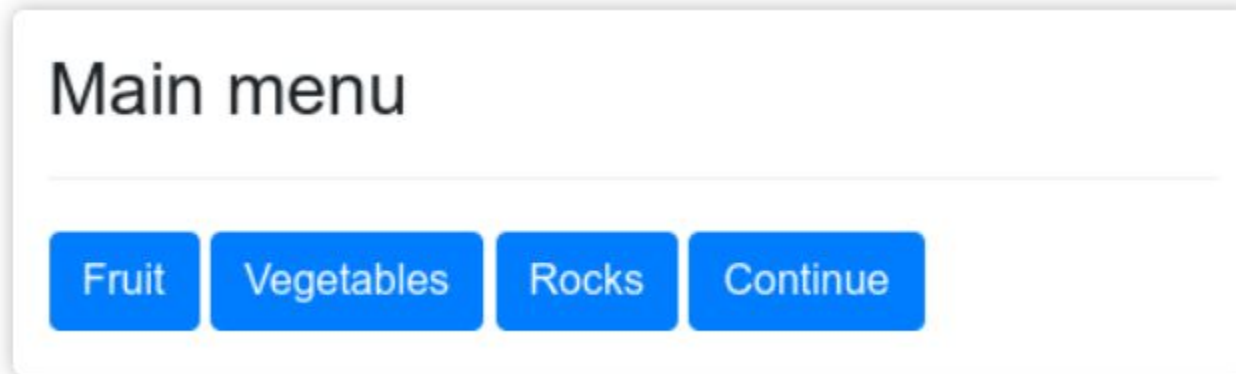
Função dispatch()

Objetivo: fornece lógica para que uma entrevista possa apresentar um sistema de menus dentro de uma tela.

Parâmetro: nome da variável do menu

Retorno: executa um dos itens do menu

Se você quiser que sair da função dispatch(), inclua uma opção *Null*. Exemplo:



Funções de data

Funções month_of(), day_of(), year_of(), and dow_of()

Objetivo: retornar uma parte da data, ano, mês, dia ou dia da semana.

Parâmetro: uma data

Retorno: parte da data de acordo com a função escolhida

Give me a date.

Date *

Continue

Funções de data

Função `format_date()`

Objetivo: retornar a data de acordo com o formato escolhido

Parâmetro: texto que contém uma data, um objeto [DADateTime](#), um objeto [datetime.datetime](#) ou um objeto [datetime.time](#)

Retorno: data formatada

A formatação da data é fornecida pelo pacote [babel.dates](#), que oferece suporte para variação de idioma e localidade.

Funções de data

Função format_date()

- `format_date(today())` returns the current date, formatted like `January 1, 2018`.
- `format_date("10/31/2016")` returns `October 31, 2016`.
- `format_date("2016-04-01")` returns `April 1, 2016`.
- `format_date('April 5, 2014')` returns `April 5, 2014`.
- `format_date('April 5, 2014', format='long')` returns `April 5, 2014`.
- `format_date('April 5, 2014', format='full')` returns `Saturday, April 5, 2014`.
- `format_date('April 5, 2014', format='EEEE')` returns `Saturday`.
- `format_date('April 5, 2014', format='yyyy')` returns `2014`.
- `format_date('April 5, 2014', format='yy')` returns `14`.
- `format_date('April 5, 2014', format='MMMM')` returns `April`.
- `format_date('April 5, 2014', format='MMM')` returns `Apr`.

Funções de data

Função format_time()

Objetivo: o mesmo da função format_date(), só que retorna a hora de acordo com o formato escolhido

Parâmetro: texto que contém uma hora, um objeto [DADateTime](#), um objeto [datetime.datetime](#) ou um objeto [datetime.time](#)

Retorno: hora formatada

A formatação da hora também é fornecida pelo pacote [babel.dates](#).

Funções de data

Função format_time()

- `format_time(current_datetime())` returns the current time, formatted like `12:00 AM`.
- `format_time("04:01:23")` returns `4:00 AM`.
- `format_time("04:01:23", format='h')` returns `4`.
- `format_time("04:01:23", format='hh')` returns `04`.
- `format_time("04:01:23", format='m')` returns `1`.
- `format_time("04:01:23", format='mm')` returns `01`.
- `format_time("04:01:23", format='ss')` returns `23`.
- `format_time("04:01:23", format='a')` returns `AM`.
- `format_time("04:01:23", format='z')` returns `EST`, or whatever the current time zone is.

Funções de data

Função format_datetime()

Objetivo: é uma combinação das funções format_date() e format_time()

Parâmetro: texto que contém uma data e hora, um objeto [DADateTime](#), um objeto [datetime.datetime](#) ou um objeto [datetime.time](#)

Retorno: data e hora formatadas

- `format_datetime('4/5/2014 07:30')` returns April 5, 2014 at 07:30:00 AM EST.
- `format_datetime('4/5/2014 07:30', 'h:mm in MMMM')` returns '7:30 in April'.

Funções de data

Função today()

Objetivo: exibir a data atual

Parâmetro: não possui

Parâmetro opcional: timezone

Retorno: data atual de acordo com o timezone

```
question: |  
    Today's date is ${ today() }.  
mandatory: True
```

Funções de data

Função current_datetime()

Objetivo: exibir a data e hora atuais

Parâmetro: não possui

Parâmetro opcional: timezone

Retorno: data e hora atuais de acordo com o timezone

```
question: |  
    % if current_datetime() < as_datetime('October 14, 1066'):  
    The Normans have not yet invaded  
    England.  
    % else:  
    Unfortunately, the Normans have  
    already invaded England.  
    % endif  
mandatory: True
```

Funções de data

Função date_difference()

Objetivo: calcular a diferença entre duas datas no formato escolhido, anos, meses, dias, minutos, etc

Parâmetro: data inicial e data final

Retorno: intervalo que deve ser formatado para a exibição

The length of time between my birthday and Christmas can be described several ways.

- 31.720238095238095 weeks
- 222.04166666666666 days
- 5329.0 hours
- 319740.0 minutes
- 19184400 seconds

Funções de data

Função date_interval()

Objetivo: é usada para fazer cálculos de intervalos de data

Parâmetro: formato e valor do intervalo

Retorno: intervalo no formato escolhido

Valores permitidos para o intervalo: anos, meses, dias, semanas, horas, minutos, segundos e microssegundos.

```
question: |  
    In three days, it will be  
    ${ format_date( current_datetime() + date_interval(days=3) ) }.  
mandatory: True
```

Funções de texto

Funções bold(), and italic()

Objetivo: são úteis na inclusão variáveis em um documento

Parâmetro: texto

Retorno: texto formatado

```
subquestion: |  
  * Your phone number: ${ bold(phone_number) }  
  * Your fax number: ${ bold(fax_number) }
```

- Your phone number: **202-555-2030**
- Your fax number:

Funções de texto

Função single_paragraph()

Objetivo: retira as quebras de linha

Parâmetro: texto

Retorno: texto em um único parágrafo

```
---  
question: Summary of your answers  
subquestion: |  
    When I asked you the meaning of life, you said:  
  
    > ${ single_paragraph(meaning_of_life) }  
  
field: ok_to_proceed  
---
```

Outros tipos de função

Função qr_code()

Objetivo: permite incluir a instrução de QR code usando Python.

Parâmetro: texto

Retorno: imagem do QR code



Outros tipos de função

Função send_email()

Objetivo: enviar e-mail para algum destinatário.

E-mail test

This interview will send you an e-mail.

[Continue](#)

Please fill in the following information.

Your First Name *

Your Last Name *

Your E-mail *

A Picture *

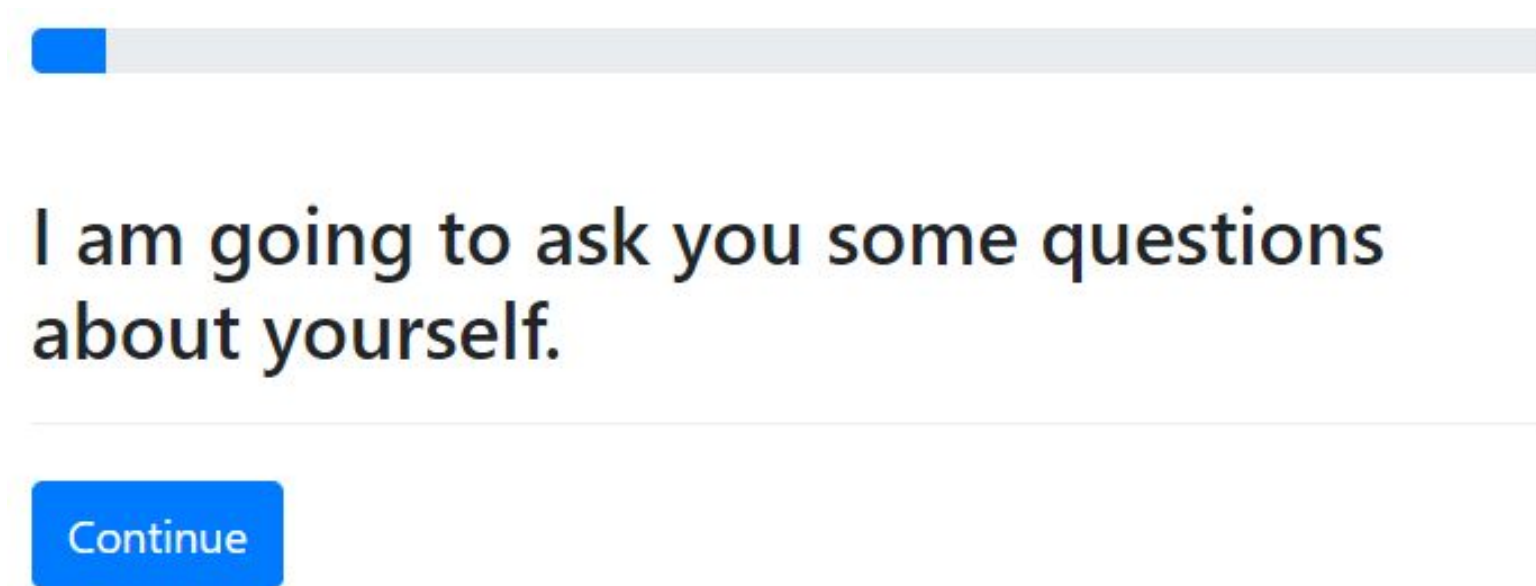
Drag & drop files here ...

[Select file...](#) [Browse ...](#)

Outros tipos de função

Funções set_progress()

Objetivo: mostra a barra de progressão durante o preenchimento do documento.



Outros tipos de função

Função showif()

Objetivo: retorna um valor baseado em uma condição

Parâmetro: nome de uma variável, valor se verdadeiro e valor se falso (opcional)

Retorno: valor da variável se a condição especificada for verdadeira e, caso contrário, retorna texto vazio ou o valor fornecido como 3º argumento.

Qual o nome do(a) primeiro(a) devedor(a)?

Nome Completo: *

Nacionalidade: *

☒ brasileiro(a)

☐ estrangeiro(a)

Profissão *

Qual o nome do(a) primeiro(a) devedor(a)?

Nome Completo: *

Nacionalidade: *

☐ brasileiro(a)

☒ estrangeiro(a)

Qual é a *
nacionalidade?

Profissão *

Muito mais!!!!

O Jinja2, linguagem usada nos templates, possui também um extenso conjunto de funções!

<https://jinja.palletsprojects.com/en/2.10.x/templates/#list-of-builtin-filters>



Atividades

Dia 08 – Atividades



Breaking the Law

Copyright 2019 Roberto Novaes (www.robertonovaes.com.br)

Copyright 2019 SiLEX Sistemas (www.silexsistemas.com.br)

Atribuição-NãoComercial-CompartilhaIgual 4.0 Internacional



Realização:



U F *m* G

Apoio:

MARCELO CANDIOTTO **MCSA**
Sociedade de Advogados



ASSOCIAÇÃO
BRASILEIRA DE
**LAWTECHS &
LEGALTECHS**