

# **Estruturas de arquivos**

*Release 0.1*

**Jander Moreira**

31 mar. 2022



|          |                          |          |
|----------|--------------------------|----------|
| <b>1</b> | <b>Módulo campo</b>      | <b>1</b> |
|          | Índice de Módulos Python | 5        |
|          | Index                    | 7        |



---

## Módulo campo

---

campo

**class** `estruturarq.campo.campo_comum.CampoBasico` ( *tipo: str* )

Estruturação básica do campo como menor unidade de informação.

**abstract bytes\_para\_valor** ( *dado: bytes* )

Conversão de uma sequência de bytes para armazenamento no valor do campo, de acordo com a representação de dados :param dado: sequência de bytes :return: o valor do campo de acordo com seu tipo

**comprimento** ( )

Obtém o comprimento do campo, se ele for fixo :return: o comprimento do campo se for fixo ou None se

for variável

**comprimento\_fixo** ( )

Retorna se o comprimento é ou não fixo :return: True para comprimento fixo ou False para variável

**copy** ( )

Cópia “rasa” deste campo :return: outra instância com os mesmos valores

**escreva** ( *arquivo: BinaryIO* )

Conversão do valor para sequência de bytes e armazenamento no arquivo :param arquivo: arquivo binário aberto com permissão de escrita

**leia** ( *arquivo: BinaryIO* )

Conversão dos dados lidos para o valor do campo, obedecendo à organização e formato de representação :param arquivo: arquivo binário aberto com permissão de leitura

**abstract property valor**

Recuperação, com as devidas conversões, do atributo `__valor` :return: o valor de `__valor`

**abstract valor\_para\_bytes** ( ) → *bytes*

Conversão do valor do campo para sequência de bytes de acordo com a representação de dados :return:

**class** `estruturarq.campo.campo_comum.CampoBruto` ( *valor=""* )

Implementação das funções de um campo bruto, ou seja, sem organização de campo. O valor é sempre armazenado como cadeia de caracteres.

**bytes\_para\_valor** ( *dado: bytes* )

Conversão de sequência de bytes para valor o campo, considerando uma cadeia de caracteres simples :param dado: a sequência de bytes

**property valor**

Recuperação, com as devidas conversões, do atributo `__valor` :return: o valor de `__valor`

**valor\_para\_bytes ( ) → bytes**

Conversão do valor do campo (cadeia de caracteres) para uma sequência de bytes, usando codificação UTF-8 :return: a sequência de bytes

**class** `estrutarq.campo.campo_cadeia.CampoCadeiaBasico ( tipo: str, valor: str = '' )`

Classe básica para cadeias de caracteres

**bytes\_para\_valor ( dado: *bytes* )**

Atribuição de valor a partir da representação de dados :param dado: sequência de bytes

**property valor**

Recuperação, com as devidas conversões, do atributo `__valor` :return: o valor de `__valor`

**valor\_para\_bytes ( ) → bytes**

Conversão da cadeia de caracteres para sequência de bytes com codificação UTF-8 :return: sequência de bytes

**class** `estrutarq.campo.campo_cadeia.CampoCadeiaFixo ( comprimento: int, **kwargs )`

Classe para cadeia de caracteres com comprimento\_bloco fixo e preenchimento de dados inválidos

**class** `estrutarq.campo.campo_cadeia.CampoCadeiaPrefixado ( *args, **kwargs )`

Classe para cadeia de caracteres prefixada pelo comprimento\_bloco

**class** `estrutarq.campo.campo_cadeia.CampoCadeiaTerminador ( **kwargs )`

Classe para cadeia de caracteres com terminador

**class** `estrutarq.campo.campo_inteiro.CampoIntBasico ( tipo: str, valor: int = 0 )`

Classe básica para campo inteiro

**bytes\_para\_valor ( dado: *bytes* )**

Conversão de uma sequência de bytes (representação textual) para inteiro :param dado: sequência de bytes

**property valor: int**

Recuperação, com as devidas conversões, do atributo `__valor` :return: o valor de `__valor`

**valor\_para\_bytes ( ) → bytes**

Conversão do valor inteiro para sequência de bytes usando representação textual e codificação UTF-8 :return: sequência de bytes

**class** `estrutarq.campo.campo_inteiro.CampoIntBinario ( **kwargs )`

Classe para inteiro em formato binário (big endian) com 8 bytes e complemento para 2 para valores negativos

**bytes\_para\_valor ( dado: *bytes* )**

Conversão de uma sequência de bytes (binária big-endian com sinal) para inteiro :param dado: sequência de bytes

**valor\_para\_bytes ( ) → bytes**

Conversão do valor inteiro para sequência de bytes usando representação binária big-endian com sinal :return: sequência de bytes

**class** `estrutarq.campo.campo_inteiro.CampoIntFixo ( comprimento: int, **kwargs )`

Classe para inteiro textual com tamanho fixo

```
class estrutarq.campo.campo_inteiro.CampoIntPrefixado ( **kwargs )
    Classe para inteiro textual com prefixo de comprimento_bloco

class estrutarq.campo.campo_inteiro.CampoIntTerminador ( terminador: bytes = b'\x00',
**kwargs )
    Classe para inteiro textual com terminador

class estrutarq.campo.campo_real.CampoRealBasico ( tipo: str, valor: float = 0 )
    Classe básica para campo real

    bytes_para_valor ( dado: bytes )
        Conversão de sequência de bytes com valor textual para valor real :param dado: sequência
        de 8 bytes

    property valor: float
        Recuperação, com as devidas conversões, do atributo __valor :return: o valor de __valor

    valor_para_bytes ( ) → bytes
        Conversão do valor do campo para sequência de bytes textual :return: a sequência de bytes
        no padrão especificado

class estrutarq.campo.campo_real.CampoRealBinario ( **kwargs )
    Classe para real em formato binário usando IEEE 754 de precisão dupla

    bytes_para_valor ( dado: bytes )
        Conversão de sequência de bytes com representação IEEE 754 de precisão dupla para real
        :param dado: sequência de 8 bytes

    valor_para_bytes ( ) → bytes
        Conversão do valor do campo para sequência de bytes no padrão IEEE 754 de precisão dupla
        :return: a sequência de bytes no padrão especificado

class estrutarq.campo.campo_real.CampoRealFixo ( comprimento: int, **kwargs )
    Classe para campo real com representação textual de comprimento_bloco fixo

class estrutarq.campo.campo_real.CampoRealPrefixado ( **kwargs )
    Classe para campo real com representação textual de comprimento_bloco fixo

class estrutarq.campo.campo_real.CampoRealTerminador ( terminador: bytes = b'\x00',
**kwargs )
    Classe para campo real com representação textual de comprimento_bloco fixo

class estrutarq.campo.campo_tempo.CampoDataBinario ( **kwargs )
    Classe para armazenamento de data (dia, mês e ano) para armazenamento em formato binário.

class estrutarq.campo.campo_tempo.CampoDataFixo ( **kwargs )
    Classe para data, em número de segundos desde 1/1/1970, 0h00min00s usando armazenamento
    em cadeia de caracteres no formato 'formato_data'.

class estrutarq.campo.campo_tempo.CampoHoraBinario ( **kwargs )
    Classe para horário usando armazenamento em valor inteiro em binário, com sinal, big-endian.

class estrutarq.campo.campo_tempo.CampoHoraFixo ( **kwargs )
    Classe horário usando armazenamento em cadeia de caracteres no formato 'formato_hora'.

class estrutarq.campo.campo_tempo.CampoTempoBasico ( tipo: str, formato: str, apenas_data:
bool, valor: str = '', **kwargs )
    Classe básica para campo de tempo (data + horário), armazenado internamente como o número
```

de segundos desde 1/1/1970, 0h00min00s.

Quando apenas a data é armazenada, o horário é ajustado para 12h00min00s, para evitar problemas com fuso horário.

**property valor:** `str`

Recuperação, com as devidas conversões, do atributo `__valor` :return: o valor de `__valor`

**class** `estrutarq.campo.campo_tempo.CampoTempoBasicoBinario ( *args, **kwargs )`

Implementação das conversões tempo-> binário e binário->tempo

**bytes\_para\_valor** ( *dado: bytes* )

Conversão da representação binária (8 bytes, big-endian, com sinal) para valor inteiro de segundos :param dado: bytes da representação do inteiro em binário

**valor\_para\_bytes** ( ) → `bytes`

Conversão do valor do tempo em segundos para representação em inteiro binário (8 bytes, big-endian, com sinal) :return: a sequência de bytes

**class** `estrutarq.campo.campo_tempo.CampoTempoBasicoFixo ( *args, **kwargs )`

Implementação das conversões tempo-> binário e binário->tempo

**bytes\_para\_valor** ( *dado: bytes* )

Conversão da representação binária (8 bytes, big-endian, com sinal) para valor inteiro de segundos :param dado: bytes da representação do inteiro em binário

**valor\_para\_bytes** ( ) → `bytes`

Conversão do valor do tempo em segundos para representação em inteiro binário (8 bytes, big-endian, com sinal) :return: a sequência de bytes

**class** `estrutarq.campo.campo_tempo.CampoTempoBinario ( **kwargs )`

Classe para tempo (data + horário), em número de segundos desde 1/1/1970, 0h00min00s usando armazenamento em valor inteiro em binário, com sinal, big-endian.

**class** `estrutarq.campo.campo_tempo.CampoTempoFixo ( **kwargs )`

Classe para tempo (data + horário), em número de segundos desde 1/1/1970, 0h00min00s usando armazenamento em cadeia de caracteres no formato 'formato\_tempo'.



## e

estrutarq  
    estrutarq.campo, 1  
    estrutarq.campo.campo\_cadeia, 2  
    estrutarq.campo.campo\_comum, 1  
    estrutarq.campo.campo\_inteiro, 2  
    estrutarq.campo.campo\_real, 3  
    estrutarq.campo.campo\_tempo, 3



**B**

bytes\_para\_valor() (método estruturarq.campo.-campo\_cadeia.CampoCadeiaBasico), 2  
bytes\_para\_valor() (método estruturarq.campo.-campo\_comum.CampoBasico), 1  
bytes\_para\_valor() (método estruturarq.campo.-campo\_comum.CampoBruto), 1  
bytes\_para\_valor() (método estruturarq.campo.-campo\_inteiro.CampoIntBasico), 2  
bytes\_para\_valor() (método estruturarq.campo.-campo\_inteiro.CampoIntBinario), 2  
bytes\_para\_valor() (método estruturarq.campo.-campo\_real.CampoRealBasico), 3  
bytes\_para\_valor() (método estruturarq.campo.-campo\_real.CampoRealBinario), 3  
bytes\_para\_valor() (método estruturarq.campo.-campo\_tempo.CampoTempoBasicoBinario), 4  
bytes\_para\_valor() (método estruturarq.campo.-campo\_tempo.CampoTempoBasicoFixo), 4

**C**

CampoBasico (classe em estruturarq.campo.campo\_comum), 1  
CampoBruto (classe em estruturarq.campo.campo\_comum), 1  
CampoCadeiaBasico (classe em estruturarq.campo.campo\_cadeia), 2  
CampoCadeiaFixo (classe em estruturarq.campo.-campo\_cadeia), 2  
CampoCadeiaPrefixado (classe em estruturarq.-campo.campo\_cadeia), 2  
CampoCadeiaTerminador (classe em estruturarq.-campo.campo\_cadeia), 2  
CampoDataBinario (classe em estruturarq.campo.campo\_tempo), 3  
CampoDataFixo (classe em estruturarq.campo.-campo\_tempo), 3  
CampoHoraBinario (classe em estruturarq.campo.campo\_tempo), 3

CampoHoraFixo (classe em estruturarq.campo.-campo\_tempo), 3  
CampoIntBasico (classe em estruturarq.campo.-campo\_inteiro), 2  
CampoIntBinario (classe em estruturarq.campo.-campo\_inteiro), 2  
CampoIntFixo (classe em estruturarq.campo.campo\_inteiro), 2  
CampoIntPrefixado (classe em estruturarq.campo.campo\_inteiro), 3  
CampoIntTerminador (classe em estruturarq.campo.campo\_inteiro), 3  
CampoRealBasico (classe em estruturarq.campo.-campo\_real), 3  
CampoRealBinario (classe em estruturarq.campo.-campo\_real), 3  
CampoRealFixo (classe em estruturarq.campo.-campo\_real), 3  
CampoRealPrefixado (classe em estruturarq.campo.campo\_real), 3  
CampoRealTerminador (classe em estruturarq.-campo.campo\_real), 3  
CampoTempoBasico (classe em estruturarq.campo.campo\_tempo), 3  
CampoTempoBasicoBinario (classe em estruturarq.campo.campo\_tempo), 4  
CampoTempoBasicoFixo (classe em estruturarq.-campo.campo\_tempo), 4  
CampoTempoBinario (classe em estruturarq.campo.campo\_tempo), 4  
CampoTempoFixo (classe em estruturarq.campo.-campo\_tempo), 4  
comprimento() (método estruturarq.campo.campo\_comum.CampoBasico), 1  
comprimento\_fixo() (método estruturarq.campo.-campo\_comum.CampoBasico), 1  
copy() (método estruturarq.campo.campo\_comum.CampoBasico), 1

**E**

escreva() (método estruturarq.campo.campo\_comum.CampoBasico), 1

estrutarq.campo  
módulo, 1

estrutarq.campo.campo\_cadeia  
módulo, 2

estrutarq.campo.campo\_comum  
módulo, 1

estrutarq.campo.campo\_inteiro  
módulo, 2

estrutarq.campo.campo\_real  
módulo, 3

estrutarq.campo.campo\_tempo  
módulo, 3

campo\_tempo.CampoTempoBasicoFixo),  
4

## L

leia() (método estrutarq.campo.campo\_comum.CampoBasico), 1

## M

módulo

estrutarq.campo, 1

estrutarq.campo.campo\_cadeia, 2

estrutarq.campo.campo\_comum, 1

estrutarq.campo.campo\_inteiro, 2

estrutarq.campo.campo\_real, 3

estrutarq.campo.campo\_tempo, 3

## V

valor (propriedade estrutarq.campo.campo\_cadeia.CampoCadeiaBasico ), 2

valor (propriedade estrutarq.campo.campo\_comum.CampoBasico ), 1

valor (propriedade estrutarq.campo.campo\_comum.CampoBruto ), 2

valor (propriedade estrutarq.campo.campo\_inteiro.CampoIntBasico ), 2

valor (propriedade estrutarq.campo.campo\_real.CampoRealBasico ), 3

valor (propriedade estrutarq.campo.campo\_tempo.CampoTempoBasico ), 4

valor\_para\_bytes() (método estrutarq.campo.campo\_cadeia.CampoCadeiaBasico), 2

valor\_para\_bytes() (método estrutarq.campo.campo\_comum.CampoBasico), 1

valor\_para\_bytes() (método estrutarq.campo.campo\_comum.CampoBruto), 2

valor\_para\_bytes() (método estrutarq.campo.campo\_inteiro.CampoIntBasico), 2

valor\_para\_bytes() (método estrutarq.campo.campo\_inteiro.CampoIntBinario), 2

valor\_para\_bytes() (método estrutarq.campo.campo\_real.CampoRealBasico), 3

valor\_para\_bytes() (método estrutarq.campo.campo\_real.CampoRealBinario), 3

valor\_para\_bytes() (método estrutarq.campo.campo\_tempo.CampoTempoBasicoBinario),  
4

valor\_para\_bytes() (método estrutarq.campo.-