Limpeza de dados

Usando métodos básicos para limpar os dados

Limpeza de dados

Usando métodos básicos para limpar os dados

Quando escrevemos nossos programas, normalmente manipulamos excessões conhecidas, desta forma, evitamos um “crash” no sistema.

Quando realizamos Web Scraping, também precisamos praticar uma codificação defensiva para manipular o inesperado.

Limpeza de dados

Usando métodos básicos para limpar os dados

Realizar Web Scraping nem sempre é como nós esperamos, usar fontes de dados conhecidas, onde os dados estão bem formatados e onde podemos descartar dados que não nos atendem nem sempre é possível.

Pontuação errada, caixa alta, quebras de linha, grafia incorreta, os dados “sujos” podem ser um problema para o scraping.

Limpeza de dados

Usando métodos básicos para limpar os dados

Vamos fazer um exemplo de scraping pegando o conteúdo do site **https://pt.wikipedia.org/wiki/Python**

```
from urllib.request import urlopen
from bs4 import BeautifulSoup

html = urlopen("https://pt.wikipedia.org/wiki/Python")
bsObj = BeautifulSoup(html)
conteudo = bsObj.find("div", {"id":"mw-content-text"}).get_text()
conteudo = conteudo.split(" ")
print(conteudo)
```

Limpeza de dados

Usando métodos básicos para limpar os dados

Veja parte do resultado. Precisamos fazer uma limpeza para poder utilizar estes dados.

['\xa0Nota:', 'Para', 'outros', 'significados,', 'veja', 'Python',
'(desambiguação).\n\n\nPython\n\n\n\n\n\nParadigma\n\nMultiparadigma:Orientação', 'a', 'objetosProgramação', 'imperativaProgramação',
'funcional\n\n\nSurgido', 'em\n\n1991', '(27–28', 'anos)[1]\n\n\nÚltima', 'versão\n\n3.7.0', '(27', 'de', 'junho', 'de', '2018;', 'há', '11',
'meses[2])\n\n\nCriado', 'por\n\nGuido', 'van', 'Rossum[1]\n\n\nEstilo', 'de', 'tipagem:\n\nDinâmica,', 'forte\n\n\nInfluenciada',
'por\n\nABC,[3]', 'ALGOL', '68,', 'C[3],', 'Haskell,', 'Icon,' 'ISBN\xa0978-1-4493-4037-7\xa0\nLigações', 'externas[editar', '|', 'editar', 'código-
fonte]\n\n\nOutros', 'projetos', 'Wikimedia', 'também', 'contêm', 'material', 'sobre', 'este', 'tema:\n\n\n\n\nDefinições', 'no',
'Wikcionário\n\n\n\n\nLivros', 'e', 'manuais', 'no', 'Wikilivros\n\n\n\n\nCitações', 'no', 'Wikiquote\n\n\n\n\nCategoria', 'no',
'Commons\n\n\nCommons\nWikiquote\nWikilivros\nWikcionário\n\nSítio', 'oficial', '(em', 'inglês)\nPython', 'no', 'GitHub\nWiki', 'da',
'comunidade', 'brasileira', 'de', 'usuários\nSite', 'da', 'comunidade', 'portuguesa', 'de', 'usuários\nPython', 'no', 'DMOZ\n\nv\xa0•\xa0e',
'Linguagens', 'de', 'programação', '\n\nEsotéricas\xa0·', 'Comparação\xa0·', 'História\xa0·', 'Programa', 'Olá', 'Mundo\xa0·', 'Algoritmo', 'de',
'Trabb', 'Pardo-Knuth\n\nAssembly', '(asm)\nAngelScript\nC\nC++\nC#\nFortran\nGo\nGroovy\nHaskell\nJava\nJavaScript',
'(JS)\nKotlin\nLisp\nLua\nObjective-C\nOCaml\nPerl\nPHP\nPython\nR\nRuby\nRust\nScala\nShell\nSwift\nTypeScript\nVisual', 'Basic',
'.NET', '(VB.NET)\nmais...\n\n', 'Categoria\xa0·', '', 'Lista\n\n\n\n\n']

Limpeza de dados

Usando métodos básicos para limpar os dados

Vamos criar um método chamado limpar_texto para realizar a limpeza dos caracteres não desejados.

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
import re
import string

html = urlopen("https://pt.wikipedia.org/wiki/Python")
bsObj = BeautifulSoup(html)
conteudo = bsObj.find("div", {"id":"mw-content-text"}).get_text()
conteudo = limpar_texto(conteudo)
print(conteudo)
```

Limpeza de dados

Usando métodos básicos para limpar os dados

Implementação do método limpar_texto.

```
def limpar_texto(texto):
    texto = texto.strip()
    texto_limpo = []
    # Trocando um ou mais caracteres de nova linha (enter) por um espaço.
    texto = re.sub("\n+", " ", texto)
    # Trocando um ou mais espaços por um espaço.
    texto = re.sub(" +", " ", texto)
    # Remover os caracteres de controle
    texto = texto.replace(u'\xa0', u'')
    # Remover números entre colchetes (citações Wikipedia)
    texto = re.sub("\[[0-9]*\]", "", texto)
    texto = texto.split(" ")
    for item in texto:
        item = item.strip()
        # string.punctuation == '!"#$%&\'()*+,-./:;<=>?@[\\]^_`{|}~'
        # Removendo caracteres de pontuação antes e depois da string
        item = item.strip(string.punctuation)
        if len(item) > 1 or (item.lower() == 'a' or item.lower() == 'e' or item.lower() == 'o'):
            texto_limpo.append(item)

    return texto_limpo
```

Limpeza de dados

Usando métodos básicos para limpar os dados

Veja parte do resultado. Usando o método limpar_texto.

```
['Nota', 'Para', 'outros', 'significados', 'veja', 'Python', 'desambiguação', 'Python', 'Paradigma',
'Multiparadigma:Orientação', 'a', 'objetosProgramação', 'imperativaProgramação', 'funcional', 'Surgido', 'em',
'1991', '27–28', 'anos', 'Última', 'versão', '3.7.2', '24', 'de', 'dezembro', 'de', '2018', 'há', '11', 'meses', 'Criado',
'por', 'Guido', 'van', 'Rossum', 'Estilo', 'de', 'tipagem', 'Dinâmica', 'forte', 'Influenciada', 'por', 'ABC', 'ALGOL', '68',
'Haskell', 'Icon', 'Java', 'Lisp', 'Modula-3', 'Perl', 'Smalltalk', 'Influenciou', 'Boo', 'Falcon', 'Fantom', 'Groovy',
'JavaScript', 'Nimrod', 'Py', 'Ruby', 'Squirrel', 'Swift', 'Principais', 'implementações', 'CPython', 'IronPython',
'Jython', 'PyPy', 'Extensão', 'do', 'arquivo', 'py', 'pyc', 'pyd', 'pyo', 'pyw', 'pyz', 'Página', 'oficial', 'www.python.org',
'Python', 'uma', 'linguagem', 'usuários', 'Site', 'da', 'comunidade', 'portuguesa', 'de', 'usuários', 'Python', 'no',
'DMOZ', 'Linguagens', 'de', 'programação', 'Esotéricas·', 'Comparação·', 'História·', 'Programa', 'Olá', 'Mundo·',
'Algoritmo', 'de', 'Trabb', 'Pardo-Knuth', 'Assembly', 'asm', 'Fortran', 'Go', 'Groovy', 'Haskell', 'Java', 'JavaScript', 'JS',
'Kotlin', 'Lisp', 'Lua', 'Objective-C', 'OCaml', 'Perl', 'PHP', 'Python', 'Ruby', 'Rust', 'Scala', 'Shell', 'Swift', 'TypeScript',
'Visual', 'Basic', 'NET', 'VB.NET', 'mais', 'Categoria·', 'Lista']
```

FIM