baby_names

July 15, 2021

Script for the “Baby Names exercise” of Google’s Python Class https://developers.google.com/edu/python/exercises/baby-names Licensed under the Apache License, Version 2.0 http://www.apache.org/licenses/LICENSE-2.0

The program takes as input a html file containing names of (male and female) babies born in a given year, and order them alphabetically.

[7]:

```
import sys
import re
```

The function extract_names takes a html file and returns a list containing the year, the names of babies ordered alphabetically, and the popularity index of the names:

[8]:

```
def extract_names(filename):
  file = open(filename,'r')
  filedata = file.read()
  file.close()
  match = re.search(r'Popularity in \d+', filedata)
  year = match.group()[-4:]
  names = [year]
  components = re.findall(r'(\d+?)</td><td>(\w+?)</td><td>(\w+?)<', filedata)
  for component in components:
    names.append(component[1] + ' ' + component[0])
    names.append(component[2] + ' ' + component[0])
  names = sorted(names)
  return names
```

main() stops and return a usage message if no arguments are given:

[9]:

```
def main():
  args = sys.argv[1:]
  if not args:
    sys.exit('usage: [--summary] file [file ...]')

  # Notices the summary flag and removes it from args if it is present:
  summary = False
  if args[0] == '--summary':
    summary = True
    del args[0]
  for filename in args:
    names = extract_names(filename)

    # Makes text separated by line breaks out of the whole list:
    text = '\n'.join(names)

    # If the option summary is entered, creates a file filename.summary containing the text:
    if summary:
      outf = open(filename + '.summary', 'w')
      outf.write(text)
      outf.close()
    else:
      print(text)
```

[11]:

```
if __name__ == '__main__' and '__file__' in globals():
  main()
```

[ ]: