

Apr 04, 11 10:19

FileVisit.py

Page 1/1

```

'''
Created on Mar 29, 2011

@author: ola
'''
5  import os

10 def bigfiles(dirname,min_size):
    large = []
    #print dirname
    for sub in os.listdir(dirname):
        path = os.path.join(dirname,sub)
15         if os.path.isdir(path):
            large.extend(bigfiles(path,min_size))
        else:
            size = os.path.getsize(path)
            if size > min_size:
20                 large.append((path,size))
    return large

25 def ftree(indent,dirname,depth):
    sub_depth = indent.count("+")
    if sub_depth >= depth:
        return
    print indent,dirname
    contents = os.listdir(dirname)
30     for sub in contents:
        path = os.path.join(dirname,sub)
        if os.path.isdir(path):
            ftree(indent+"---+",path,depth)
        else:
35             print indent,path

if __name__ == "__main__":
    #ftree("+", "/Users/ola/Desktop", 3)
    bigs = bigfiles("/Users/ola/Desktop/courses/6python", 10000)
40     for f in bigs:
        print f

```

Apr 04, 11 10:19

Koch.py

Page 1/1

```

'''
Created on Apr 4, 2011

@author: ola
'''
5  import turtle

def draw(itors):
    flake = "FRFRF"
10     for steps in range(itors):
        flake = flake.replace("F", "FLFRFLF")

    for move in flake:
        if move == "F":
            turtle.forward(100.0/3**(itors-1))
15         elif move == "L":
            turtle.left(60)
        elif move == "R":
            turtle.right(120)

20 draw(2)
x = raw_input()

```

Apr 04, 11 10:19

## Merge.py

Page 1/1

```

'''
Created on Dec 1, 2010

@author: ola
'''
5
import random

def merge(left, right):
10
    result = []
    i, j = 0, 0
    while i < len(left) and j < len(right):
        if left[i] <= right[j]:
            result.append(left[i])
            i += 1
15
        else:
            result.append(right[j])
            j += 1
    result.extend(left[i:])
    result.extend(right[j:])
20
    return result

def mergesort(list):
25
    if len(list) < 2:
        return list
    else:
        middle = len(list) / 2
        left = mergesort(list[:middle])
        right = mergesort(list[middle:])
30
        return merge(left, right)

def isSorted(lst):
35
    for i, n in enumerate(lst[1:]):
        if n < lst[i]:
            return False
    return True

40
nums = [x for x in range(0,100)]
random.Shuffle(nums)
snums = mergesort(nums)

print isSorted(nums)
45
print isSorted(snums)

print nums
print snums

```

Apr 04, 11 10:19

## Sierpinski1.py

Page 1/1

```

'''
Created on Apr 3, 2011

@author: ola
'''
5
"""
create Sierpinski Triangle using PIL
"""

10
import Image      # PIL
import ImageDraw  # PIL
import random

def midpoint(p,q):
15
    return ((p[0]+q[0])/2.0, (p[1]+q[1])/2.0)

def sierpinski(p1,p2,p3, image, level):
    """
    calculates points for sub triangles, uses recursion for steps
20
    """

    if level <= 0:
        return

25
    # draw triangles each step through

    image.line([p1,p2])
    image.line([p2,p3])
    image.line([p1,p3])

30
    # make smaller triangles from each point and adjacent midpoints

    sierpinski(p1,midpoint(p1,p2),midpoint(p1,p3),image,level-1)
    sierpinski(p2,midpoint(p2,p3),midpoint(p2,p1),image,level-1)
35
    sierpinski(p3,midpoint(p3,p2),midpoint(p3,p1),image,level-1)

def chaos(p1,p2,p3,image,itters):
    plist = [p1,p2,p3]
    current = (random.uniform(p1[0],p2[0]),.2)
    for x in xrange(itters):
        vert = random.choice(plist)
        mid = midpoint(current,vert)
        image.point(mid,fill='black')
45
        current = mid

    # starting point for equilateral triangle
    tpoints = ((0, 700), (700, 700), (350, 0))

50
    level = 8
    size = tpoints[1]
    picture = Image.new('RGB', size,color='blue')
    picture2 = Image.new('RGB',size,color='white')
55
    drawnimage = ImageDraw.Draw(picture)
    drawnimage2 = ImageDraw.Draw(picture2)

    # draw the triangle and calculate next triangle corner coordinates
    sierpinski(tpoints[0],tpoints[1],tpoints[2],drawnimage, level)
60
    picture.show()

    chaos(tpoints[0],tpoints[1],tpoints[2],drawnimage2,100000)
    picture2.show()

```

Apr 04, 11 10:19

SimpleGrammar.py

Page 1/1

```

'''
Created on Apr 4, 2011

@author: ola
5 '''

import random

def create_content():
10 """
    return a dictionary used for generating random sentences
    """
    adjectives = ["<color>", "slimy", "wonderful", "beautiful", "obese", "teeny", "<adj>", "<adj>"]
    colors = ["green", "red", "yellow", "blue", "maroon"]
15 rules = {"<color>": colors, "<adj>": adjectives}

    return rules

20 def expand(sentence, rules):
    """
    expand sentence using rules as source of meta-words
    """
    sent = ""
25     for w in sentence.split():
        if w.startswith("<"):
            chosen = random.choice(rules[w])
            sent += expand(chosen, rules) + " "
        else:
30             sent += w + " "
    return sent.strip()

def create():
    rules = create_content()
35     print expand("the <adj> dog ate the <adj> bone", rules)
    print expand("the <color> house was a <adj> edifice", rules)

if __name__ == "__main__":
    create()

```