```
import os

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

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

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

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

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

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

if __name__ == "__main__":
    create()