# -\*- coding: utf-8 -\*-  
  
# Аттрибуты могут иметь не только объекты, но и их классы  
  
from random import randint, choice  
  
  
# посчитаем леммингов  
class Lemming:  
 pass  
  
  
total\_lemmings = 0  
lemming\_1 = Lemming()  
total\_lemmings += 1  
lemming\_2 = Lemming()  
total\_lemmings += 1  
lemming\_3 = Lemming()  
total\_lemmings += 1  
  
family = []  
family\_size = randint(16, 32)  
while len(family) < family\_size:  
 new\_lemming = Lemming()  
 family.append(new\_lemming)  
 total\_lemmings += 1  
print(total\_lemmings)  
  
  
# пусть сам класс следит за количеством своих объектов  
class Lemming:  
 # можно определять атрибуты на уровне класса, тогда они "привязаны" к классу  
 total = 0  
  
 def \_\_init\_\_(self):  
 # обращаться - через именование класса  
 Lemming.total += 1  
  
  
family = []  
family\_size = randint(16, 32)  
while len(family) < family\_size:  
 new\_lemming = Lemming()  
 family.append(new\_lemming)  
print(Lemming.total)  
  
  
# или даже  
burrow = []  
burrow\_depth = randint(90, 100)  
while len(burrow) < burrow\_depth:  
 family = []  
 family\_size = randint(16, 32)  
 while len(family) < family\_size:  
 new\_lemming = Lemming()  
 family.append(new\_lemming)  
 burrow.append(family)  
print(Lemming.total)  
print(burrow)  
  
  
# в аттрибутах класса допустимо любое выражение пайтона  
class Lemming:  
 total, names = 0, ['Peter', 'Anna', 'Nik', 'Sofi', 'Den', 'Lora', 'Bred', ]  
 names\_count = len(names)  
 some\_text = 'Варкалось, хливкие шорьки пырялись по наве...'  
 some\_var = some\_text + names[-1]  
  
 def \_\_init\_\_(self):  
 Lemming.total += 1  
 self.name = choice(Lemming.names)  
  
 def \_\_str\_\_(self):  
 return 'Lemming ' + self.name  
  
 def check\_class\_attrs(self):  
 print('Lemming.total', Lemming.total)  
 print('Lemming.names', Lemming.names)  
 print('Lemming.names\_count', Lemming.names\_count)  
 print('Lemming.some\_text', Lemming.some\_text)  
 print('Lemming.some\_var', Lemming.some\_var)  
  
  
new\_lemming = Lemming()  
print('Lemming.total', Lemming.total)  
print('Lemming.names', Lemming.names)  
print('Lemming.names\_count', Lemming.names\_count)  
print('Lemming.some\_text', Lemming.some\_text)  
print('Lemming.some\_var', Lemming.some\_var)  
new\_lemming.check\_class\_attrs()