# -\*- coding: utf-8 -\*-  
  
from random import randint  
  
  
# Реализуем модель человека.  
# Человек может есть, работать, играть, ходить в магазин.  
# У человека есть степень сытости, немного еды и денег.  
# Если сытость < 0 единиц, человек умирает.  
# Человеку надо прожить 365 дней.  
from termcolor import cprint  
  
  
class Man:  
  
 def \_\_init\_\_(self, name):  
 self.name = name  
 self.fullness = 50  
 self.house = None  
  
 def \_\_str\_\_(self):  
 return 'Я - {}, сытость {}'.format(  
 self.name, self.fullness)  
  
 def eat(self):  
 if self.house.food >= 10:  
 cprint('{} поел'.format(self.name), color='yellow')  
 self.fullness += 10  
 self.house.food -= 10  
 else:  
 cprint('{} нет еды'.format(self.name), color='red')  
  
 def work(self):  
 cprint('{} сходил на работу'.format(self.name), color='blue')  
 self.house.money += 50  
 self.fullness -= 10  
  
 def watch\_MTV(self):  
 cprint('{} смотрел MTV целый день'.format(self.name), color='green')  
 self.fullness -= 10  
  
 def shopping(self):  
 if self.house.money >= 50:  
 cprint('{} сходил в магазин за едой'.format(self.name), color='magenta')  
 self.house.money -= 50  
 self.house.food += 50  
 else:  
 cprint('{} деньги кончились!'.format(self.name), color='red')  
  
 def go\_to\_the\_house(self, house):  
 self.house = house  
 self.fullness -= 10  
 cprint('{} Вьехал в дом'.format(self.name), color='cyan')  
  
 def act(self):  
 if self.fullness <= 0:  
 cprint('{} умер...'.format(self.name), color='red')  
 return  
 dice = randint(1, 6)  
 if self.fullness < 20:  
 self.eat()  
 elif self.house.food < 10:  
 self.shopping()  
 elif self.house.money < 50:  
 self.work()  
 elif dice == 1:  
 self.work()  
 elif dice == 2:  
 self.eat()  
 else:  
 self.watch\_MTV()  
  
  
class House:  
  
 def \_\_init\_\_(self):  
 self.food = 50  
 self.money = 0  
  
 def \_\_str\_\_(self):  
 return 'В доме еды осталось {}, денег осталось {}'.format(  
 self.food, self.money)  
  
  
citizens = [  
 Man(name='Бивис'),  
 Man(name='Батхед'),  
 Man(name='Кенни'),  
]  
  
  
my\_sweet\_home = House()  
for citisen in citizens:  
 citisen.go\_to\_the\_house(house=my\_sweet\_home)  
  
for day in range(1, 366):  
 print('================ день {} =================='.format(day))  
 for citisen in citizens:  
 citisen.act()  
 print('--- в конце дня ---')  
 for citisen in citizens:  
 print(citisen)  
 print(my\_sweet\_home)  
  
# Создадим двух людей, живущих в одном доме - Бивиса и Батхеда  
# Нужен класс Дом, в нем должн быть холодильник с едой и тумбочка с деньгами  
# Еда пусть хранится в холодильнике в доме, а деньги - в тумбочке.