

## **Разработка программы автоматического генерирования билетов для контрольных на язык Python**

*Кизянов Антон Олегович*

*Приамурский государственный университет имени Шолом-Алейхема  
студент*

### **Аннотация**

В данной статье продемонстрирован процесс создания программы генерирования билетов для контрольных.

**Ключевые слова:** Python, random.

## **Development of automatic ticket generation program for the control in the Python language**

*Kizyanov Anton Olegovic*

*Sholom-Aleichem Priamursky State University  
Student*

### **Abstract**

This article demonstrated the process of creating a program for generating the ticket control.

**Keywords:** Python, random.

Предположим, вы читаете географию группе из 35 студентов и хотите провести контрольную работу на знание столиц штатов в США. Увы, обстановка в вашем классе такая, что вы не можете быть уверены в том, что студенты не будут списывать друг у друга. Вы хотели бы составить экзаменационные билеты таким образом, чтобы вопросы в них располагались в случайном порядке, благодаря чему каждый будет отличаться от другого, и это затруднит списывание ответов. Разумеется, составлять такие билеты вручную – задача утомительная и к тому же отнимающая много времени.

Для ознакомления с языком программирования Python прочтите следующие статьи. В.А.Машков, В.И.Литвиненко рассказали о применении языка программирования python для решения задач самодиагностики на системном уровне [1]. Г.Д.Бухарова, и П.С.Комельских рассказали о важности и необходимости внедрения языка программирования Python в процесс обучения студентов [2]. Г.С.Сейдаметов продемонстрировал особенности использования языка программирования python в подготовке будущих инженеров-программистов [3]. Э.А.Усеинов продемонстрировал использование объектно-ориентированного программирования в рамках дисциплины «язык программирования python» [4].

Первый шаг состоит в том, чтобы создать «скелет» сценария и наполнить его данными билета.

```
import random

capitals = {'Alabama': 'Montgomery'
            , 'Alaska': 'Juneau'
            , 'Arizona': 'Phoenix'
            .....
            , 'Wisconsin': 'Madison'
            , 'Wyoming': 'Cheyenne'}
capitalsItems = list(capitals.items())

for quizNum in range(35):
```

Рис. 1

Поскольку эта программа должна располагать вопросы и ответы в случайном порядке, понадобится импортировать модуль `random`, чтобы использовать его функции. Переменная `capitals` содержит словарь, в котором штаты США играют роль ключей, а значениями являются названия столиц штатов. А поскольку хотите создать 35 билетов, код, который будет фактически генерировать файлы билетов и ключей ответов, должен быть помещен в цикл `for`, выполняющий 35 итераций.

Код в цикле будет повторен 35 раз – по одному разу на каждый билет, в связи с чем достаточно сосредоточивать внимание в цикле каждый раз только на одном билете. Прежде всего, необходимо создать фактический файл билета. Он должен иметь уникальное имя, а также содержать некий стандартный заголовок с пустыми полями для имени, даты и класса, которые будут заполняться студентами. Далее вам нужно будет получить список штатов, расположенных в случайном порядке, который впоследствии можно будет использовать для создания вопросов и ответов к каждому билету.

```
quizFile = open('capitalsquiz%s.txt' % (quizNum + 1), 'w')
answerKeyFile = open('capitalsquiz_answers%s.txt' % (quizNum + 1), 'w')

quizFile.write('Name:\n\nDate:\n\nPeriod:\n\n')
quizFile.write((' ' * 20) + 'State Capitals Quiz (Form %s)' % (quizNum + 1))
quizFile.write('\n\n')

states = list(capitals.keys())
random.shuffle(states)
```

Рис. 2

Файлы будут иметь имена `capitalsquiz<N>.txt` где `<N>` - это уникальный номер билета, который берется из переменного цикла `quizNum`. Ключи ответов для файлов `capitalsquiz<N>.txt` будут храниться в текстовых файлах `capitalsquiz_answers<N>.txt`. При каждом прохождении цикла вместо заместителя `%s` в строках `'capitalsquiz%s.txt'` и `'capitalsquiz_answers%s.txt'` будет подставляться значение `(quizNum + 1)`, поэтому файлами первого из создаваемых билетов и ключа ответа будут `capitalquiz1.txt` и

capitalsquiz\_answerws1.txt. Эти файлы будут создаваться вызовами функции `open()`.

Инструкция `write()` создают заголовок билета с полями, которые будут заполняться студентами. Наконец, с помощью функции `random.shuffle()`, которая случайным образом переупорядочивает список любых переданных ей значений создает рандомизированный список всех штатов США.

Теперь необходимо сгенерировать варианты ответов для каждого вопроса, предоставляя возможность выбора из ответов, обозначенных буквами от А до D. Вам понадобится создать ещё один цикл `for` – он будет генерировать содержимое для каждого из 50 вопросов билета. Далее будет третий, вложенный цикл `for`, предназначенный для генерации вариантов множественного выбора для каждого вопроса.

```
for questionNum in range(50):

    correctAnswer = capitals[states[questionNum]]
    wrongAnswers = list(capitals.values())
    del wrongAnswers[wrongAnswers.index(correctAnswer)]
    wrongAnswers = random.sample(wrongAnswers, 3)

    answerOptions = wrongAnswers + [correctAnswer]
    random.shuffle(answerOptions)
```

Рис. 3

Корректный ответ можно легко получить – он хранится в виде значения в слове `capitals`. Данный цикл итерирует по штатам, содержащимся в перемешанном списке штатов, от `states[0]` до `states[49]`, находит каждый штат в `capitals` и сохраняет название его столицы в переменной `correctAnswer`.

Со списком возможных неправильных ответов дело обстоит несколько сложнее. Вы сможете получить его, продублировав все значения из словаря `capitals`, удалив правильный ответ и выбрав три случайных значения из этого списка. Функция `random.sample()` упрощает этот выбор. Её первый аргумент – это список, из которого вы хотите выбирать значения; второй аргумент – это количество значений, которые вы хотите выбирать. Полный список вариантов ответа представляет собой сочетание трех неправильных ответов с правильным. Наконец, ответы следуют перемешать, чтобы правильный ответ не всегда соответствовал варианту D.

Теперь все, что осталось сделать, - это записать вопрос в файл билета, а ответ - в файл ключей ответов.

```
quizFile.write('%s. What is the capital of %s?\n' % (questionNum + 1, states[questionNum]))
for i in range(4):
    quizFile.write('    %s. %s\n' % ('ABCD'[i], answerOptions[i]))
quizFile.write('\n')

answerKeyFile.write('%s. %s\n' % (questionNum + 1, 'ABCD'[answerOptions.index(correctAnswer)]))
quizFile.close()
answerKeyFile.close()
```

Рис. 4

Цикл `for`, перебирает целые числа от 0 до 3, записывает варианты ответов в список `answerOptions`. В выражении `'ABCD'[i]` строка `'ABCD'` трактуется как массив с элементами `'A'`, `'B'`, `'C'` и `'D'`, выбираемыми на соответствующей итерации цикла.

В последней строке выражение `answerOptions.index(correctAnswer)` находит целочисленный индекс правильного ответа среди случайно расположенных вариантов, а вычисление выражения `'ABCD'[answerOptions.index(correctAnswer)]` дает буквенное обозначение правильного варианта ответа, подлежащего записи в файл ключа ответа.

Ниже показан примерный вид содержимого файла `capitalquiz1.txt`, хотя, разумеется, вопросы и варианты ответа в файле будут выглядеть иначе, в зависимости от результатов вызова функции `random.shuffle()`.

```
Name:
Date:
Period:

State Capitals Quiz (Form 1)

1. What is the capital of Georgia?
A. Dover
B. Santa Fe
C. Boise
D. Atlanta

2. What is the capital of Nebraska?
A. Lansing
B. Austin
C. Springfield
D. Lincoln

3. What is the capital of Kansas?
A. Phoenix
B. Dover
C. Topeka
D. Tallahassee
```

Рис. 5

```
1. D
2. D
3. C
4. C
5. B
6. A
7. D
8. A
9. C
10. B
11. C
12. A
13. D
14. A
15. C
16. A
17. C
18. A
19. D
20. A
21. A
22. B
23. C
24. B
25. B
```

Рис. 6

Полный код программы представлен ниже.

```
import random

capitals = {'Alabama': 'Montgomery',
            , 'Alaska': 'Juneau'
            , 'Arizona': 'Phoenix'
            .....
            , 'Wisconsin': 'Madison'
            , 'Wyoming': 'Cheyenne'}
capitalsItems = list(capitals.items())

for quizNum in range(35):
    quizFile = open('capitalsquiz%s.txt' % (quizNum + 1), 'w')
    answerKeyFile = open('capitalsquiz_answers%s.txt' % (quizNum + 1), 'w')

    quizFile.write('Name:\n\nDate:\n\nPeriod:\n\n')
    quizFile.write((' ' * 20) + 'State Capitals Quiz (Form %s)' % (quizNum + 1))
    quizFile.write('\n\n')

    states = list(capitals.keys())
    random.shuffle(states)

    for questionNum in range(50):

        correctAnswer = capitals[states[questionNum]]
        wrongAnswers = list(capitals.values())
        del wrongAnswers[wrongAnswers.index(correctAnswer)]
        wrongAnswers = random.sample(wrongAnswers, 3)

        answerOptions = wrongAnswers + [correctAnswer]
        random.shuffle(answerOptions)

        quizFile.write('%s. What is the capital of %s?\n' % (questionNum + 1, states[questionNum]))
        for i in range(4):
            quizFile.write('    %s. %s\n' % ('ABCD'[i], answerOptions[i]))
        quizFile.write('\n')

        answerKeyFile.write('%s. %s\n' % (questionNum + 1, 'ABCD'[answerOptions.index(correctAnswer)]))
    quizFile.close()
    answerKeyFile.close()
```

Рис. 7

Вывод: Написали программу автоматической генерации случайных билетов неограниченного количества вариантов и ключей к ним.

## Библиографический список

1. Машков В.А., Литвиненко В.И. Использование языка программирования python 3 и системы компьютерной алгебры sympy на факультативных занятиях по теории чисел // В сборнике: Электротехнические и компьютерные системы Издательство: Одесский национальный политехнический университет (Одесса) С. 48-54 [Электронный ресурс]. URL: <http://elibrary.ru/item.asp?id=23422667> (дата обращения: 25.01.2017)
2. Бухарова Г.Д., Комельских П.С. Важность и необходимость внедрения языка программирования python в процесс обучения студентов // В сборнике: новые информационные технологии в образовании Материалы VII международной научно-практической конференции. Российский государственный профессионально-педагогический университет. 2014 Издательство: Российский государственный профессионально-педагогический университет (Екатеринбург) С. 40-42. [Электронный ресурс]. URL: <http://elibrary.ru/item.asp?id=22278620> (дата обращения: 25.01.2017)

3. Сейдаметов Г.С. Особенности использования языка программирования python в подготовке будущих инженеров-программистов // В сборнике: INTERNATIONAL SCIENTIFIC REVIEW Издательство: Олимп (Иваново) С. 50-51 [Электронный ресурс]. URL: <http://elibrary.ru/item.asp?id=24983350> (дата обращения: 25.01.2017)
4. Усеинов Э.А. Объектно-ориентированное программирование в рамках дисциплины «язык программирования python» // В сборнике: ученые записки крымского инженерно-педагогического университета Издательство: Государственное бюджетное образовательное учреждение высшего образования Республики Крым «Крымский инженерно-педагогический университет» (Симферополь) С. 157-160. [Электронный ресурс]. URL: <http://elibrary.ru/item.asp?id=24836776> (дата обращения: 25.01.2017)