УДК 004.42

Мирошниченко Дарья Вячеславовна

АВТОМАТИЗАЦИЯ РАБОЧЕГО ПРОЦЕССА: ПЕРЕВОД ТЕСТА В ШАБЛОН С ПОМОЩЬЮ PYTHON

Автоматизация процессов является актуальной темой на протяжении многих лет. Компании стараются автоматизировать работу сотрудников, чтобы увеличить объемы и качество работы, что позволяет сократить время и трудовые ресурсы компании.Цель данной статьи рассмотреть разработанный код, который автоматизирует процесс перевод теста для электронного курса в шаблон для дальнейшего импорта в систему Moodle.Разработанный код позволяет автоматизировать работу разработчика электронного курса и сократить время импорта вопросов в банк заданий в несколько раз.

Импорт вопросов, автоматизация процессов, электронный курс, программный код.

Miroshnichenko Daria Vyacheslavovna

WORKFLOW AUTOMATION: TRANSLATING TEXT INTO A TEMPLATE USING PYTHON

Process automation has been a hot topic for many years. Companies are trying to automate the work of employees in order to increase the volume and quality of work, which reduces the time and labor resources of the company.The purpose of this article is to review the developed code that automates the process of converting an e-course test into a template for further import into the Moodle system.The developed code allows you to automate the work of the developer of the electronic course and reduce the time of importing questions into the task bank several times.

Import of questions, automation of processes, electronic course, program code.

Введение

На данный момент существует проблема при формировании электронного банка тестов по дисциплинам в системе, разработанной с помощью системы управления образовательными электронными ресурсами, которая заключается в том, что перевод теста в шаблон для импорта в банк заданий производится вручную, что занимает большое количество времени [1].

Цель данной статьи рассмотреть разработанный код, который автоматизирует процесс перевод теста для электронного курса в шаблон для дальнейшего импорта в систему Moodle.

Основная часть

Разработанный код работает с текстовыми документами и ориентируется по основным патернам, которые присущи каждому тесту, так как оформление его стандартное и обязательное для всех. Данный код ускоряет процесс в 5 раз, что упрощает работу с ресурсом и позволяет разработчику сконцентрировать свое внимание на качестве создаваемых вопросов.

На рис. 1 продемонстрирован изначальный вид вопросов закрытого типа, которые необходимо обработать.

1. Инструмент, используемый для создания исходного текста программы, называется:
А) **Редактор**
Б) Компилятор/ассемблер
В) Программный симулятор
Г) Аппаратный эмулятор
Д) Программатор

2. При выборе возможных архитектур и характеристик микроконтроллеров рекомендуется ориентироваться на:
А) Использование узкоспециализированных, мелкосерийных вариантов;
Б) **Использование наиболее свежих моделей от разработчиков**;
В) Использование давно выпускаемых моделей, серийный выпуск которых завершается;
Г) Использование уже известных моделей.

Рис. 1. Пример изначального вида закрытых вопросов

Пример кода

Рассмотрим программный код, написанный на языке программирования Python. Данный код переводит вопросы закрытого типа из файла формата Word в блокнот с расширением txt для дальнейшего импорта в систему. Рассмотрим данный код.

1. Импортируем модуль для роботов с файлами формата .docx, и создаем функцию, которая принимает путь к файлу в качестве аргумента. Далее создается пустая строка для хранения результирующего текста [2, 3].

```
import docx

# Открытие файла
def process_text(file_path):
    doc = docx.Document(file_path)
    result = ""
    for paragraph in doc.paragraphs:
        text = paragraph.text.strip()
```

2. Далее проверяется текст на начилие начала строки с чисел от 1 до 70 (количество в тексте), если оно встречается, то преобразуется в формате «::1.::». Далее добавляется символ «{» в новой строке.

```
# Проверка на номер вопроса
        if any(word.endswith(".") and word[:-1].isdigit() and 1 <= int(word[:-1]) <= 70 for word in paragraph.text.split()):

# Заменяем число на формат "::1.::"
            text += "\n"
            text = " ".join("::" + word[:-1] + ".::" if word.endswith(".") and word[:-1].isdigit() and 1 <= int(word[:-1]) <= 70 else word for word in paragraph.text.split())
            text += "\n"
            text += "{"
```

3. Далее идет проверка на шрифт, если текст полужирный, то в начале строки ставится символ «=», в остальных случаях – «~». Если строка пустая, то ставится символ «}».

```
elif any(word.endswith(".") and word[:-1].isdigit() and 1 <= int(word[:-1]) <= 100 for word in paragraph.text.split()):
    # Заменяем число на формат "::1.::"
            text += "\n"
            text = " ".join("::" + word[:-1] + ".::" if word.endswith(".") and word[:-1].isdigit() and 1 <= int(word[:-1]) <= 100 else word for word in paragraph.text.split() )
            text += "\n"
```

```
        text += "{"

    elif paragraph.runs and paragraph.runs[0].bold:
        text = "=" + " " + text[0:]
    elif paragraph.runs:
        text = "~" + " " + text[0:]
    else:
        text += "}" + "\n"
    result += text + "\n"
```

4. Когда обработка текста закончена выводится результат в командную строку. Результат работы кода представлен на рис. 2.

```
::1::: Инструмент, используемый для создания исходного текста программы, называется:
{
   Редактор
   Компилятор/ассемблер
   Программный симулятор
   Аппаратный эмулятор
   Программатор
}

::2::: При выборе возможных архитектур и характеристик микроконтроллеров не рекомендуется ориентироваться на
{
   Использование узкоспециализированных, мелкосерийных вариантов;
   Использование наиболее свежих моделей от разработчиков;
   Использование давно выпускаемых моделей, серийный выпуск которых завершается;
   Использование уже известных моделей.
   Ничего из перечисленного
}
```

Рис. 2. Итог обработки закрытых вопросов по шаблону

На рис. 3 продемонстрирован изначальный вид вопросов открытого типа, которые необходимо обработать.

42. Операторы языка программирования, предназначенные для организации многократного исполнения некоторых фрагментов программы называются операторами
(цикла)

43. Операторы языка программирования, предназначенные для осуществления ветвлений в зависимости от результатов вычислений называются операторами
(условными)

Рис. 3. Пример изначального вида открытых вопросов

Пример кода

Рассмотрим программный код, написанный на языке программирования Python. Данный код переводит вопросы открытого типа из файла формата Word в блокнот с расширением txt для дальнейшего импорта в систему. Данный код отличается от предыдущего тем, что вместо проверки текста на полужирный и обычный, производится обработка полужирного текста в скобках. Если программа его находит, то добавляет в начало строки символ «=».

```
for paragraph in doc.paragraphs:
    # Поиск текста в скобках и его замена на "{\n= текст \n}"
    regex_bold = r'\((.*?)\)'
    bold_matches = re.findall(regex_bold, paragraph.text)
    for bold_text in bold_matches:
        paragraph.text = paragraph.text.replace(f"({bold_text})", f"\n{{\n=
{bold_text} \n}}")
```

Результат работы кода продемонстрирован на рис. 4.

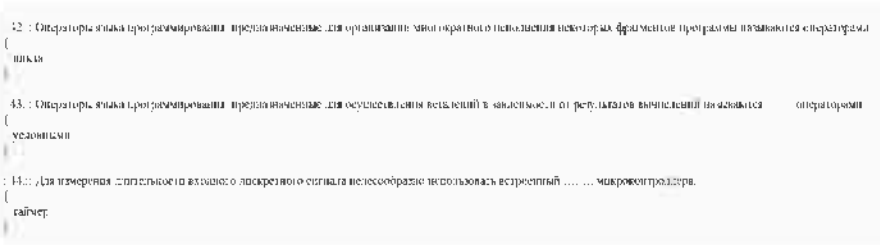

Рис. 4. Итог обработки открытых вопросов по шаблону

Выводы

Разработанный код позволяет автоматизировать работу разработчика электронного курса и сократить время импорта вопросов в банк заданий в несколько раз. Описанный в статье способ является самым простым и удобным способом, разработанным на языке Python

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Официальная документация Moodle URL: https://moodle.org/?lang=ru (дата обращения: 15.04.2024).

2. Файлы в python, ввод-вывод URL: https://pythonru.com/osnovy/fajly-v-python-vvod-vyvod (дата обращения: 25.04.2024).
3. Working with Documents – Python .docx URL: https://www.geeksforgeeks.o
4. rg/working-with-documents-python-docx-module/ (дата обращения: 25.04.2024).

Мирошниченко Дарья Вячеславовна, студентка Донского государственного технического университета, Россия, город Ростов-на-Дону, площадь Гагарина 1, 344018, телефон: +7(988)552-45-61, email: my_sun_md@mail.ru.

Miroshnichenko Daria Vyacheslavovna, student of the Don State Technical University, Russia, Rostov-on-Don, Gagarin Square 1, 344018, phone: +7(988)552-45-61, email: my_sun_md@mail.ru.