

## Лабораторная работа № 4

### Программная обработка текстовых и табличных документов

**Задача:** Предполагая, что отдел кадров принял на работу 5 сотрудников, составил трудовые договоры, и на их основе подготовил документ **Список сотрудников.xlsx** (структура в таблице 1), подготовить автоматическое заполнение приказов о приеме на работу каждого сотрудника и частичного заполнения личной карточки сотрудника, опираясь на данные из документов **Штат сотрудников.txt** и **Список сотрудников.xlsx**.

#### *Порядок выполнения задания:*

- I. Составить табличный документ **Список сотрудников (текущая дата).xlsx**, содержащий данные не менее чем о 5 работниках, имеющий следующую структуру:

Таблица 1– Структура документа «Список сотрудников»

Табельный номер	Фамилия	Имя	Отчество	Код структурного подразделения	Должность	Дата трудового договора	№ трудового договора
9(3)	A(20)	A(15)	A(15)	9(2)	X(20)	9(2).9(2).9(4)	9(4)

#### **Примечания:**

- Пока не используем возможности базы данных, табельный номер сотрудника и номер трудового договора назначить самостоятельно, контролировать, чтобы они не повторялись.
  - С табличным документом можно работать в программе LibreOffice.Calc, но сохранять документ нужно в формате .xlsx!
  - При вводе данных код подразделения и должность следует согласовывать со штатным расписанием.
- II. Создать шаблоны для документов **Приказ о приеме на работу** и **Личная карточка**, сохранить их в папке **Шаблоны** в формате .docx. Определить поля для программного заполнения и вставить в них переменные (рисунки 1-2). Реквизиты организации должны быть введены в шаблоны документов.

		Форма по ОКУД	0301001	
???		по ОКПО	???	
наименование организации				
		Номер документа	Дата составления	
ПРИКАЗ			{{Date}}	
(распоряжение) о приеме работника на работу				
			Дата	
Принять на работу		с	{{Date}}	
		по		
{{Second name}} {{First name}} {{Middle name}} фамилия, имя, отчество			Табельный номер {{Person number}}	
В	{{Struct_subdiv}} структурное подразделение			
{{Post_sotr}} должность (специальность, профессия), разряд, класс (категория) квалификации				
условия приема на работу, характер работы				
с тарифной ставкой (окладом)	{{Salary}} цифрами	руб.		коп.
надбавкой	{{Premium}} цифрами	руб.		коп.
с испытанием на срок				месяца (ев)
Основание:				
Трудовой договор от	" {{d}} "	{{md}}	20 {{y}} г. № {{N_contract}}	
Руководитель организации	{{post_supervisor}} должность		{{iof_supervisor}} личная подпись	{{iof_supervisor}} расшифровка подписи
С приказом (распоряжением) работник ознакомлен			" " 20 г.	
		личная подпись		

Рисунок 1 – Шаблон приказа о приеме на работу

						Форма по ОКУД	0301002
						по ОКПО	???
(наименование организации)							
Дата составления	Табельный номер	Идентификационный номер налогоплательщика	Номер страхового свидетельства государственного пенсионного страхования	Алфавит	Характер работы	Вид работы (основная, по совместительству)	Пол (мужской, женский)
{{Date}}	{{Person number}}						

## ЛИЧНАЯ КАРТОЧКА

### работника

#### I. ОБЩИЕ СВЕДЕНИЯ

Трудовой договор		номер	{{N_contract}}
		дата	{{Date}}
1. Фамилия	{{Second name}}		Имя
			{{First name}}
	Отчество		{{Middle name}}

#### II. СВЕДЕНИЯ О ВОИНСКОМ УЧЕТЕ

1. Категория запаса		6. Наименование военного комиссариата по месту жительства	
2. Воинское звание			
3. Состав (профиль)		7. Состоит на воинском учете:	
4. Полное кодовое обозначение ВУС		а) <u>общем</u> (номер команды, партии)	
5. Категория годности к военной службе		б) <u>специальном</u>	
		8.	
(отметка о снятии с воинского учета)			
Работник кадровой службы	{{post_OK}}		{{iof_OK}}
	(должность)	(личная подпись)	(расшифровка подписи)
Работник			
	(личная подпись)		
«	»	20	г.

Рисунок 2 – Шаблон личной карточки

III. Программно заполнить Приказы о приеме на работу для всех сотрудников из списка (должно получиться 5 файлов!). Имя файла, формируйте по шаблону «Приказ о приеме на работу № табеля.docx».

IV. Одновременно с приказом сформировать Личную карточку работника, куда переносятся поля: табельный номер, фамилия, имя, отчество, дата трудового договора, номер трудового договора. Также программно заполнить поля: дата составления (текущая), должность ОК, ИОФ\_ОК.

1) Добавить в проект (созданный в лабораторной работе 2) новый файл **Recruitment.py** с содержимым:

```
import openpyxl
import datetime
from docxtpl import DocxTemplate

class Recruitment:
    """
    Класс для заполнения документов
    Приказ о приеме на работу и Личная карточка
    на основе текстовых документов и
    табличного документа Список сотрудников.
    """
```

```

def save_data(self, event):
    # Метод для сохранения данных на основе шаблонов
    # и файлов с данными.
    wb = ... # Открыть "рабочую книгу"
               # "Список сотрудников.xlsx"

    # Получить список имен всех листов.
    sheets_name = wb.sheetnames
    # Получить ссылку на первый лист.
    sheet1 = wb[sheets_name[0]]
    max_row = ... # Номер максимальной строки.

    today = ... # Получить текущую дату

    date = today.strftime("%d.%m.%Y")
    # Открыть файл "Структурные подразделения.txt"
    # и сформировать словарь dict sp
    # см. предыдущую работу

    for i in range(2, max_row + 1):
        # Сформировать словарь для замены данных в файлах .docx.
        self.__context = {}
        # Определить данные для "переменных" из файлов-шаблонов.
        # Значение табельно номера получить
        # из первого столбца документа "Список сотрудников.xlsx".
        self.__context["Person_number"] =
str(sheet1.cell(row=i, column=1).value)
        # Затем аналогично задать значения остальных переменных.
        self.__context["Second_name"] = ...

        # Для переменных "Salary" (оклад) и "Premium" (надбавка)
        # данные нужно взять из файла Штат сотрудников.txt

        # Для остальных данных, например, "iof supervisor"
        # значения задать константами.

        # Вызов методов для заполнения и сохранения документов.
        self.__order()

def __order(self):
    # Открытие шаблона документа prikaz o prieme.docx для
    # редактирования с помощью библиотеки docxtpl.
    # см. предыдущую работу
    doc = ...
    doc.render(self.__context)
    # Сохранить готовый документ
    # в папке Документы (имя документа описано в п. III)

def __personal_card(self):
    # Открытие шаблона «Личной карточки»
    # Содержимое аналогично предыдущему методу.

```

- 2) Добавить собственный код в тех местах, где это необходимо (задания в подчеркнутых комментариях).
- 3) В метод **Main.py** добавить следующий код (перед командой `root.mainloop()`)

```
recr = tk.Button(root, text="Заполнить приказ и личную карточку")
recr.place(x=85, y=150)
recruitment = Recruitment()
recr.bind("<Button - 1>", recruitment.save_data)
```

**Примечание:** два знака подчеркивания перед именем означает, что метод (или поле) будет скрытым (недоступным вне класса). Таким образом поддерживается инкапсуляция объекта.