

Лабораторная работа

Генерация html-страниц на основе шаблонов и информации из базы данных

Задание 1. Вывести информацию из таблиц базы данных «Библиотека» в виде HTML-страницы. Результат показан на рисунке 1 (часть данных на рисунке обрезана).

publisher		genre		reader		author		book_author		
publisher_id	publisher_name	genre_id	genre_name	reader_id	reader_name	author_id	author_name	book_author_id	book_id	author_id
1	ЭКСМО	1	Роман	1	Иванов М.С.	1	Пушкин А.С.	1	1	4
2	ПИТЕР	2	Приключения	2	Петров Ф.С.	2	Агата Кристи	2	1	5
3	РОСМЭН	3	Детектив	3	Федоров П.Р.	3	Жюль Верн	3	2	4
4	АЛЬФА-КНИГА	4	Лирика	4	Абрамова А.А.	4	Ильф И.А.	4	2	5
5	ДРОФА	5	Фантастика	5	Самарин С.С.	5	Петров Е.П.	5	3	6
6	АСТ	6	Фэнтези	6	Туполов И.Д.	6	Булгаков М.А.	6	4	3
7	НАУКА			7	Баранов П.В.	7	Лермонтов М.Ю.	7	5	2
						8	Стругацкий А.Н.	8	6	1
						9	Стругацкий Б.Н.	9	7	7
						10	Дойл Артур Конан	10	8	8
						11	Достоевский Ф.М.	11	8	9

book						book_reader				
book_id	title	genre_id	publisher_id	year_publication	available_numbers	book_reader_id	book_id	reader_id	borrow_date	return_date
1	Двенадцать стульев	1	3	2018	1	1	4	4	2020-09-11	2020-09-24
2	Золотой теленок	1	1	2014	3	2	12	6	2020-09-11	None
3	Мастер и Маргарита	1	5	2014	5	3	29	5	2020-09-17	2020-10-10
4	Таинственный остров	2	4	2015	0	4	27	6	2020-09-18	2020-10-14
5	Пуаро ведет следствие	3	5	2008	2	5	15	4	2020-09-18	2020-10-04

Рисунок 1. Результат генерации HTML-страницы с дампом базы данных «Библиотека»

Структура программы

Для реализации программы, которая использует информацию из базы данных и шаблон html, рекомендуется создать следующую совокупность файлов (модулей):

- управляющая программа (контроллер) – в ней будет реализовываться логика программы, выполняться обращение к базе данных и генерироваться шаблон;
- программа (модель) для реализации функций обращения к базе данных;
- шаблон html-страницы, который будет формировать программа;
- файл с результатом генерации (html-код).

На рисунке 2 показана схема взаимодействия между этими модулями на примере реализации первого задания лабораторной работы.

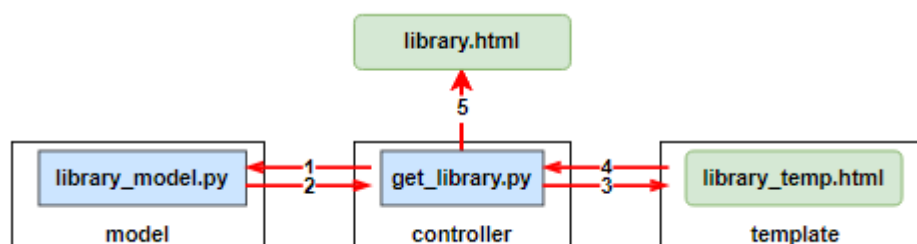


Рисунок 2. Схема взаимодействия между модулями

В программе-контроллере осуществляется:

- соединение с базой данных;
- выбор информации из базы данных;
- генерация шаблона;
- вывод результата в файл.

В **программе-модели** создаются функции, в которых выполняются запросы к базе данных и возвращается результат выборки.

В **шаблоне** создается html-страница с элементами управления шаблонизатор.

Результатом является html-страница, которую можно просмотреть в браузере.

Порядок выполнения работы

1. Для выполнения каждого задания лабораторной работы необходимо создавать отдельную папку, например, для этого задания, папка будет называться **Lab_4_1**.

2. Создать программу-контроллер `get_library.py`, в коде самостоятельно заполнить пропуски. Эта программа выводит информацию из таблицы `publisher`:

```
# импортируем необходимые модули
from jinja2 import Template
import sqlite3
from library_model import get_publisher

# устанавливаем соединение с базой данных (базу данных взять из ЛР 1)
conn = sqlite3.connect("library.sqlite")

# выбираем записи из таблицы publisher
df_publisher = get_publisher(conn)

# закрываем соединение с базой
conn.close()

# открываем шаблон из файла library_template.html и читаем информацию
f_template = open(...)
html = f_template...

f_template.close()

# создаем объект-шаблон
template = ...

# генерируем результат на основе шаблона
result_html = template.render(
    table_1 = "publisher",
    relation_1 = df_publisher,
    len = len
)

# создаем файл для HTML-страницы
f = open('library.html', 'w', encoding='utf-8-sig')
# выводим сгенерированную страницу в файл
f.write(result_html)
f.close()
```

2. Создать программу-модель `library_model.py`, в которой отобразить информацию из таблицы `publisher`.

```
import pandas as pd

def get_publisher(conn):
    return pd.read_sql("SELECT * FROM publisher", conn)
```

3. Создать шаблон library_templ.html:

```
<html>
  <head>
    <title> Дамп базы данных </title>
  </head>
  <body>
    <p>
      <b>{{table_1}} </b>
    </p>
    <!-- количество строк в таблице -->
    {% set n = len(relation_1) %}
    <table>
      <!-- формируем заголовок таблицы (имена столбцов) -->
      <tr>
        {% for name in relation_1.dtypes.index %}
          <th>{{name}}</th>
        {% endfor %}
      </tr>

      <tbody>
        <!-- для всех строк таблицы -->
        {% for i in range(n) %}
          <tr>
            <!-- выводим значение каждого столбца -->
            {% for name in relation_1.dtypes.index %}
              <td> {{ relation_1.loc[i, name] }} </td>
            {% endfor %}
          </tr>
        {% endfor %}
      </tbody>
    </table>
  </body>
</html>
```

3. Запустить программу get_library.py, посмотреть в браузере результат генерации страницы library.html в браузере (рисунок 3, а).

<p>a)</p> <table> <tr><th colspan="2">publisher</th></tr> <tr> <th>publisher_id</th><th>publisher_name</th></tr> <tr><td>1</td><td>ЭКСМО</td></tr> <tr><td>2</td><td>ПИТЕР</td></tr> <tr><td>3</td><td>РОСМЭН</td></tr> <tr><td>4</td><td>АЛЬФА-КНИГА</td></tr> <tr><td>5</td><td>ДРОФА</td></tr> <tr><td>6</td><td>АСТ</td></tr> <tr><td>7</td><td>НАУКА</td></tr> </table>	publisher		publisher_id	publisher_name	1	ЭКСМО	2	ПИТЕР	3	РОСМЭН	4	АЛЬФА-КНИГА	5	ДРОФА	6	АСТ	7	НАУКА	<table> <tr><th colspan="2">publisher</th></tr> <tr> <th>publisher_id</th><th>publisher_name</th></tr> <tr><td>1</td><td>ЭКСМО</td></tr> <tr><td>2</td><td>ПИТЕР</td></tr> <tr><td>3</td><td>РОСМЭН</td></tr> <tr><td>4</td><td>АЛЬФА-КНИГА</td></tr> <tr><td>5</td><td>ДРОФА</td></tr> <tr><td>6</td><td>АСТ</td></tr> <tr><td>7</td><td>НАУКА</td></tr> </table> <table> <tr><th colspan="2">genre</th></tr> <tr> <th>genre_id</th><th>genre_name</th></tr> <tr><td>1</td><td>Роман</td></tr> <tr><td>2</td><td>Приключения</td></tr> <tr><td>3</td><td>Детектив</td></tr> <tr><td>4</td><td>Лирика</td></tr> <tr><td>5</td><td>Фантастика</td></tr> <tr><td>6</td><td>Фэнтези</td></tr> </table> <p>б)</p>	publisher		publisher_id	publisher_name	1	ЭКСМО	2	ПИТЕР	3	РОСМЭН	4	АЛЬФА-КНИГА	5	ДРОФА	6	АСТ	7	НАУКА	genre		genre_id	genre_name	1	Роман	2	Приключения	3	Детектив	4	Лирика	5	Фантастика	6	Фэнтези
publisher																																																					
publisher_id	publisher_name																																																				
1	ЭКСМО																																																				
2	ПИТЕР																																																				
3	РОСМЭН																																																				
4	АЛЬФА-КНИГА																																																				
5	ДРОФА																																																				
6	АСТ																																																				
7	НАУКА																																																				
publisher																																																					
publisher_id	publisher_name																																																				
1	ЭКСМО																																																				
2	ПИТЕР																																																				
3	РОСМЭН																																																				
4	АЛЬФА-КНИГА																																																				
5	ДРОФА																																																				
6	АСТ																																																				
7	НАУКА																																																				
genre																																																					
genre_id	genre_name																																																				
1	Роман																																																				
2	Приключения																																																				
3	Детектив																																																				
4	Лирика																																																				
5	Фантастика																																																				
6	Фэнтези																																																				

Рисунок 3. Результат генерации html-страницы с одной и двумя таблицами

4. **Самостоятельное задание.** Вывести две таблицы (publisher и genre), результат показан на рисунке 3б.

5. **Самостоятельное задание.** Добавьте стили CSS и блоки <div>...</div> так, чтобы страница выглядела, как показано на рисунке 4 (воспользоваться стилями из предыдущей Лабораторной работы). Рекомендуется стили CSS сохранить в отдельном файле.

publisher		genre	
publisher_id	publisher_name	genre_id	genre_name
1	ЭКСМО	1	Роман
2	ПИТЕР	2	Приключения
3	РОСМЭН	3	Детектив
4	АЛЬФА-КНИГА	4	Лирика
5	ДРОФА	5	Фантастика
6	АСТ	6	Фэнтези
7	НАУКА		

Рисунок 4. Результат генерации html-страницы с CSS стилями

6. Создать макрос, который строит таблицу, если в него передано ее название и DataFrame с содержимым таблицы, поместить макрос в разделе <head>...</head> шаблона library_temp1.html:

```
{% macro render_table(table_name, relation) %}
  <p >
    <b>{{table_name}} </b>
  </p>
  {% set n = len(relation) %}
  <table>
    <tr>
      {% for name in relation.dtypes.index %}
        <th>{{name}}</th>
      {% endfor %}
    </tr>
    <tbody>

      {% for i in range(n) %}
        <tr>
          {% for name in relation.dtypes.index %}
            <td> {{ relation.loc[i, name] }} </td>
          {% endfor %}
        </tr>
      {% endfor %}
    </tbody>
  </table>
{% endmacro %}
```

7. Из HTML-кода шаблона library_temp1.html удалить описание таблиц, вместо них вызвать макрос, например:

```
{{ render_table(name_1, relation_1) }}
```

После генерации новой страницы library.html должна получиться страница, показанная на рисунке 4.

8. **Самостоятельное задание.** Вывести содержимое всех таблиц базы данных «Библиотека», результат показан на рисунке 1.

Задание 2. Сформировать HTML-страницу (рисунок 5) с информацией о том, какие книги брал некоторый читатель, дате выдачи книг и дате их возврата (если есть). Для книг указать автора или авторов. На страницу включить:

- поле со списком, в котором перечисляются все читатели, информация отображается о том читателе, который выбран в поле со списком;

- кнопку «Найти» для поиска читателя (пока она работать не будет, ее нужно просто вывести на страницу);

- таблица с информацией о книгах, которые брал читатель, в случае, если книга не сдана – вывести кнопку «Сдать» (кнопка пока работать не будет).

Читатель:

Федоров П.Р.
Иванов М.С.
Петров Ф.С.
Федоров П.Р.
Абрамова А.А.
Самарин С.С.
Туполев И.Д.
Баранов П.В.

Найти

Карточка

	Авторы	Дата_выдачи	Дата_возврата
Двенадцать	Ильф И.А., Петров Е.П.	2020-09-28	2020-10-07
Записки о Шерлоке Холмсе	Дойл Артур Конан	2020-10-09	2020-10-29
Затерянный мир	Дойл Артур Конан	2020-11-29	2020-12-21
Одноэтажная Америка	Ильф И.А., Петров Е.П.	2020-10-23	2020-10-28
Смерть на Ниле	Агата Кристи	2020-11-14	Сдать
Собачье сердце	Булгаков М.А.	2020-11-21	2020-12-12

Рисунок 5. Результат генерации html-страницы для читателя с id = 3

Порядок выполнения работы

1. Создать папку Lab_4_2.
2. Создать программу-контроллер get_reader_book.py, в коде самостоятельно заполнить пропуски:

```

from jinja2 import Template
import sqlite3
from reader_book_model import get_reader, get_book_reader

# задаем id читателя, для которого формируем страницу
reader_id = 3

# устанавливаем соединение с базой данных
conn = ...

# выбираем записи о том, какие книги брал читатель с параметром reader_id
# столбцы назвать Название, Авторы, Дата_выдачи, Дата_сдачи
df_book_reader = get_book_reader(conn, reader_id)

# выбираем записи из таблицы reader для формирования поля со списком
df_reader = get_reader(conn)

```

```

# закрываем соединение с базой
conn.close()

# открываем файл reader_book_tmpl.html и читаем шаблон
f_template = open(...)
html = ...
f_template.close()

# создаем объект-шаблон
template = ...

# генерируем результат на основе шаблона
result_html = template.render(
    reader_id = reader_id,
    combo_box = df_reader,
    book_reader = df_book_reader,
    len = len
)

# создаем файл для HTML-страницы
f = open('reader_book.html', 'w', encoding='utf-8-sig')
# выводим сгенерированную страницу в файл
f.write(result_html)
f.close()

```

3. **Самостоятельно** создать программу-модель `reader_book_model`. В ней реализовать функции, которые возвращают отобранные данные в виде `DataFrame`:

`get_reader` – выбирает всех читателей библиотеки;

`get_book_reader` – выбирает книги, которые брал заданный читатель (его `id` передается в качестве параметра).

4. Создать шаблон `reader_book_tmpl.html`:

```

<html>
<head>
    <title> Карточка читателя </title>

    {% macro render_table(table_name, relation) %}
    ...
    {% endmacro %}
</head>

<body>
    <p>
        <form>
            Читатель:
            <!-- Создаем поле со списком -->
            <select name="reader">
                {% set n = len(combo_box) %}
                <!-- формируем строки списка (каждое значение поля
                     reader_name - строка списка) в виде
                     <option [selected] value = код_читателя> ФИО </option>
                -->
                {% for i in range(n) %}
                    {% set id_r = combo_box.loc[i, "reader_id"] %}
                    <!-- атрибут value элемента будет совпадать с id читателя -->
                    {% if id_r == reader_id %}
                        <!-- если код читателя из таблицы совпадает с кодом
                             текущего читателя устанавливаем атрибут selected -->

```

```

        <option selected value={{id_r}}>
    {% else %}
        <option value={{id_r}}>
    {% endif %}
    {{ combo_box.loc[i, "reader_name"] }}
</option>
{% endfor %}
</select>

<input type="submit" value="Найти">
</form>
</p>
<!-- вызвать макрос render_table с параметрами
"Карточка" и DataFrame book_reader
-->
</body>
</html>

```

5. Страница reader_book.html в браузере должна выглядеть, как показано на рисунке 3 (только вместо кнопки «Сдать» в таблице будет выведено None).

Самостоятельные задания

1. Сгенерируйте страницу для другого читателя. Просмотрите результат в браузере.
2. Внесите изменения в макрос, который используется для вывода таблицы так, чтобы вместо None выводилась кнопка «Сдать». Должен получиться результат как на рисунке 4.
3. Оформите формирование поля со списком в виде макроса.

Самостоятельное задание. Сформировать HTML-страницу поиска книг произвольной структуры. На странице должна быть реализована настройка поиска (возможность выбора жанров, авторов, издательств) и вывод книг, отвечающих параметрам поиска. Пример страницы с поиском приведен на рисунке 6.

Жанр

- ☒ Детектив (8)
- ☐ Лирика (6)
- ☒ Приключения (3)
- ☒ Роман (9)
- ☐ Фантастика (3)

Автор

- ☒ Агата Кристи (5)
- ☐ Булгаков М.А. (2)
- ☐ Дойл Артур Конан (4)
- ☒ Жюль Верн (2)
- ☒ Ильф И.А. (3)
- ☐ Лермонтов М.Ю. (5)
- ☒ Петров Е.П. (3)
- ☐ Пушкин А.С. (5)
- ☐ Стругацкий А.Н. (3)
- ☐ Стругацкий Б.Н. (3)

Издательство

- ☐ АЛЬФА-КНИГА (6)
- ☐ АСТ (3)
- ☐ ДРОФА (9)
- ☐ ПИТЕР (2)
- ☐ РОСМЭН (6)
- ☐ ЭКСМО (3)

Жанры: Детектив, Приключения, Роман, Авторы: Агата Кристи, Жюль Верн, Ильф И.А., Петров Е.П.,

Название: Вокруг света за 80 дней Авторы: Жюль Верн Жанр: Приключения Издательство: ДРОФА Год_издания: 2019 Количество: 5 <input type="button" value="Выбрать"/>	Название: Двенадцать стульев Авторы: Ильф И.А., Петров Е.П. Жанр: Роман Издательство: РОСМЭН Год_издания: 2018 Количество: 1 <input type="button" value="Выбрать"/>	Название: Загадочное происшествие Авторы: Агата Кристи Жанр: Детектив Издательство: ДРОФА Год_издания: 2018 Количество: 5 <input type="button" value="Выбрать"/>
Название: Золотой теленок Авторы: Ильф И.А., Петров Е.П. Жанр: Роман Издательство: ЭКСМО Год_издания: 2014 Количество: 3 <input type="button" value="Выбрать"/>	Название: Одноэтажная Америка Авторы: Ильф И.А., Петров Е.П. Жанр: Роман Издательство: ПИТЕР Год_издания: 2015 Количество: 3 <input type="button" value="Выбрать"/>	Название: Пуаро ведет следствие Авторы: Агата Кристи Жанр: Детектив Издательство: ДРОФА Год_издания: 2008 Количество: 2 <input type="button" value="Выбрать"/>
Название: Скрытый домик Авторы: Агата Кристи Жанр: Детектив Издательство: ДРОФА Год_издания: 2018 Количество: 2 <input type="button" value="Выбрать"/>	Название: Смерть на Ниле Авторы: Агата Кристи Жанр: Детектив Издательство: ЭКСМО Год_издания: 2017 Количество: 8 <input type="button" value="Выбрать"/>	Название: Таинственный остров Авторы: Жюль Верн Жанр: Приключения Издательство: АЛЬФА-КНИГА Год_издания: 2015 Количество: 0 <input type="button" value="Выбрать"/>
Название: Убийства по алфавиту Авторы: Агата Кристи Жанр: Детектив Издательство: АЛЬФА-КНИГА Год_издания: 2017 Количество: 9 <input type="button" value="Выбрать"/>		

Рисунок 6. HTML-страница для поиска книг