Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Ульяновский государственный технический университет»

Кафедра «Вычислительная техника»

**Отчет по лабораторной работе №6**

Дисциплина: «Разработка профессиональных приложений»

Web-разработка

Вариант № 24

Выполнил:

студент группы ИВТАСбд-22

Хасанов И.Н.

Проверил:

преподаватель кафедры

«Вычислительная техника»

Исхаков И.И.

Ульяновск, 2023

Задание:

Необходимо реализовать web-интерфейс для просмотра всех объектов, хранящихся в базе данных. Предметная область - из лабораторной работы 3. Для доступа к данным использовать ORM peewee. Для реализации web-части - cherrypy.

Реализация:

Класс Person является прототипом ряда таблицы.

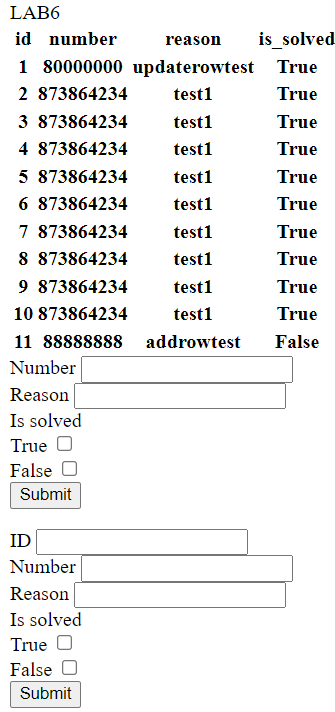
Класс Root является корнем сайта, который хранит в себе методы, которые являются страницами сайта.

Метод index является главной страницей, которая отображает таблицу БД и формы, которые позволяют модифицировать таблицу.

Метод add\_row подключается к БД и добавляет данные, взятые из формы.

Метод update\_row подключается к БД, берет данные из формы, проверяет существует ли указанная строка и вставляет данные в нее.

Пример работы:



Вывод:

При выполнении работы были изучена работа с классами в Python.

Исходный код:

from peewee import \*  
import cherrypy  
  
db = SqliteDatabase('people.db')  
  
  
class Person(Model):  
 number = IntegerField()  
 reason = CharField()  
 is\_solved = BooleanField()  
  
 class Meta:  
 database = db # модель будет использовать базу данных 'people.db'  
  
  
class Root(object):  
  
 @cherrypy.expose  
 def index(self):  
 db.connect()  
 str\_out = "<head>LAB6</head><body><table><tr><th>id</th><th>number</th><th>reason</th><th>is\_solved</th></tr><tr>"  
 for person in Person.select():  
 str\_out += '<th>' + str(person.id) + '</th><th>' + str(person.number) + '</th><th>' + person.reason \  
 + '</th><th>' + str(person.is\_solved) + '</th></tr>'  
 str\_out += """</table><form method="get" action="add\_row">  
 <label for="number">Number </label>  
 <input type="number" name="number" /><br>  
 <label for="reason">Reason </label>  
 <input type="text" name="reason" /><br>  
 <label for="status">Is solved </label><br>  
 True <input type="checkbox" name="status" value="true"/><br>  
 False <input type="checkbox" name="status" value="false"/><br>  
 <button type="submit">Submit</button>  
 </form>  
 <form method="get" action="update\_row">  
 <label for="id">ID </label>  
 <input type="number" name="id" /><br>  
 <label for="number">Number </label>  
 <input type="number" name="number" /><br>  
 <label for="reason">Reason </label>  
 <input type="text" name="reason" /><br>  
 <label for="status">Is solved </label><br>  
 True <input type="checkbox" name="status" value="true"/><br>  
 False <input type="checkbox" name="status" value="false"/><br>  
 <button type="submit">Submit</button>"""  
 db.close()  
 return str\_out  
  
 @cherrypy.expose()  
 def add\_row(self, number, reason, status):  
 db.connect()  
 b\_status = False  
 if status == "true":  
 b\_status = True  
 Person.create(number=number, reason=reason, is\_solved=b\_status)  
 db.close()  
 raise cherrypy.HTTPRedirect("/")  
  
 @cherrypy.expose()  
 def update\_row(self, id, number, reason, status):  
 db.connect()  
 b\_status = ''  
 if status == "true":  
 b\_status = True  
 elif status == "false":  
 b\_status = False  
 row\_count = Person.select().count()  
 if 0 <= int(id) <= row\_count:  
 person = Person.get(Person.id == id)  
 if number == '':  
 number = person.number  
 if reason == '':  
 reason = person.reason  
 if status == '':  
 b\_status = person.is\_solved  
 person.number = number  
 person.reason = reason  
 person.is\_solved = b\_status  
 person.save()  
 # person.update(number=number, reason=reason, is\_solved=b\_status).where(person.id == id).execute()  
 else:  
 return "NO SUCH ID"  
 db.close()  
 raise cherrypy.HTTPRedirect("/")  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 cherrypy.quickstart(Root(), '/')