Защищено: Гапанюк Ю.Е.		Демонстрация: Гапанюк Ю.Е.		
""2017	г.	""	2017 г.	
Отчет п	ю лабораторной ра	аботе № 5 п	о курсу	
	зработка интернет			
	б (количество лис	стов)		
	ИСПОЛНИТЕЛЬ: студент группы ИУ5-	51		
	Тодосиев Н.Д.	-J1	(подпись)	
	Москва, МГТУ -	- 2017		

СОДЕРЖАНИЕ

1. Описание задания лабораторной работы	. 3
2. db_script.py	. :
3. db_upd_script.py	
4. models.py	
5. views.py	
6. horse_list.html	
7. Результаты работы программы	

1. Описание задания лабораторной работы

В этой лабораторной работе вы познакомитесь с популярной СУБД MySQL, создадите свою базу данных. Также вам нужно будет дополнить свои классы предметной области, связав их с созданной базой. После этого вы создадите свои модели с помощью Django ORM, отобразите объекты из БД с помощью этих моделей и ClassBasedViews.

Для сдачи вы должны иметь:

- 1. Скрипт с подключением к БД и несколькими запросами.
- 2. Набор классов вашей предметной области с привязкой к СУБД (класс должен уметь хотя бы получать нужные записи из БД и преобразовывать их в объекты этого класса)
- 3. Модели вашей предметной области
- 4. View для отображения списка ваших сущностей

2. db_script.py

```
import MySQLdb
db = MySQLdb.connect(
  host="localhost",
  user="dbuser",
  passwd="123",
  db="first db"
c = db.cursor()
db.set character set('utf8')
c.execute('SET NAMES utf8;')
c.execute('SET CHARACTER SET utf8;')
c.execute('SET character set connection=utf8;')
c.execute("INSERT INTO 1r6 horse (horse name, horse owner, horse club) VALUES (%s,
%s, %s);", ('Каспер', 'Толстой', 'Грива и копыта'))
c.execute("UPDATE 1r6 horse SET horse owner = 'Бунин' WHERE horse owner =
'Толстой';")
db.commit()
c.execute("SELECT * FROM horse;")
entries = c.fetchall()
for entry in entries:
  print(entry)
c.close()
db.close()
```

3. db_upd_script.py

```
import MySQLdb

class Connection:
    def __init__(self, user, password, db, host='localhost'):
        self.user = user
        self.host = host
        self.password = password
        self.db = db
        self. connection = None
```

```
@property
    def connection(self):
        return self. connection
    def enter (self):
        self.connect()
    def exit (self, exc type, exc val, exc tb):
        self.disconnect()
    def connect(self):
        # открытие соединения
        if not self._connection:
            self. connection = MySQLdb.connect(
               host=self.host,
               user=self.user,
               passwd=self.password,
                db=self.db
            )
   def disconnect(self):
        # закрытие соединения
        if self. connection:
            self. connection.close()
class 1r6 horse:
          _init__(self, db_connection, horse_name, horse_owner, horse club):
        self.db connection = db connection.connection
        self.horse name = horse name
        self.horse owner = horse owner
        self.horse club = horse club
   def save(self):
        c = self.db connection.cursor()
        self.db connection.set character set('utf8')
        c.execute('SET NAMES utf8;')
        c.execute('SET CHARACTER SET utf8;')
        c.execute('SET character set connection=utf8;')
       c.execute("INSERT INTO lr6 horse (horse name, horse owner, horse club)
VALUES (%s, %s, %s);",
                  (self.horse name, self.horse owner, self.horse club))
        self.db connection.commit()
        c.close()
con = Connection("dbuser", "123", "first db")
with con:
   horse = lr6 horse(con, "Звездочка", "Галкин", "МГТУ им. Баумана")
   horse = lr6 horse(con, "Радуга", "Ужбанов", "Чистое небо")
   horse.save()
   horse = lr6 horse(con, "Маргарита", "Ихимов", "Шустрые ангелы")
   horse.save()
```

4. models.py

from django.db import models

```
# Create your models here.
```

```
class Horse(models.Model):
    horse_id=models.AutoField(primary_key=True)
    horse_name=models.TextField(max_length=20)
    horse_owner=models.TextField(max_length=80)
    horse_club=models.TextField(max_length=30)
```

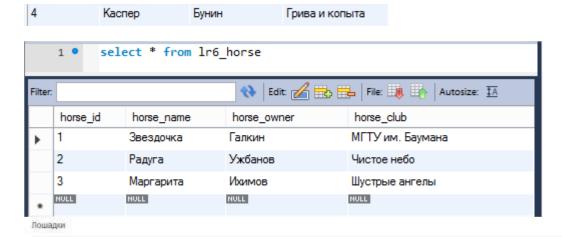
5. views.py

```
from django.shortcuts import render
from django.views.generic.list import ListView
from .models import Horse
# Create your views here.
class HorseListView(ListView):
    model = Horse

def get_context_data(self, **kwargs):
    context = super(HorseListView, self).get_context_data(**kwargs)
    return context
```

6. horse_list.html

7. Результаты работы программы



- Кличка: Звездочка с хозяином Галкин, состоящим в клубе МГТУ им. Баумана
- Кличка: Радуга с хозяином Ужбанов, состоящим в клубе Чистое небо
- Кличка: Маркарита с хозянном Ихимов, состоящим в клубе Шустрые ангелы

Лабораторная работа Nv6