

Демонстрация ЛР:
Гапанюк. Ю.Е.

**Отчет по лабораторной работе № 6 по курсу
РИП**

"Работа с СУБД"

ИСПОЛНИТЕЛЬ:

студент группы **ИУ5-54**

Наседкин И.А.

Задание и порядок выполнения

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

Текст программы

views.py

```
from django.shortcuts import render
from django.http import HttpResponse
from django.views.generic import ListView
from django.views.generic import View
from polls.models import TransactionModel
from django.views.generic import TemplateView
from polls.models import *

# Create your views here.
class AboutView(TemplateView):
    template_name = "about.html"

class TransactionListView(ListView):
    def get(self, request):
        transactions = TransactionModel.objects.all ()
        return render (request, 'transactions.html', {'transactions':
transactions})
```

urls.py

```
"""untitled1 URL Configuration

The `urlpatterns` list routes URLs to views. For more information please see:
    https://docs.djangoproject.com/en/1.10/topics/http/urls/
Examples:
Function views
    1. Add an import:  from my_app import views
    2. Add a URL to urlpatterns:  url(r'^$', views.home, name='home')
Class-based views
    1. Add an import:  from other_app.views import Home
    2. Add a URL to urlpatterns:  url(r'^$', Home.as_view(), name='home')
Including another URLconf
    1. Import the include() function: from django.conf.urls import url, include
    2. Add a URL to urlpatterns:  url(r'^blog/', include('blog.urls'))
"""
from django.conf.urls import url
from django.contrib import admin
```

```

from polls.views import TransactionListView
from polls.views import AboutView
from polls.views import *

urlpatterns = [
    url(r'^admin/', admin.site.urls),
    url(r'^transactions/$', TransactionListView.as_view()),
    url(r'^about/$', AboutView.as_view()),
]

```

models.py

```

from django.db import models

# Create your models here.
class UserModel(models.Model):
    id = models.IntegerField(primary_key=True)
    name = models.CharField(max_length=30)
    email = models.CharField(max_length=30)
    phone = models.CharField(max_length=20)

class BankModel(models.Model):
    idbank = models.IntegerField(primary_key=True)
    director = models.CharField(max_length=30)
    address = models.CharField(max_length=255)

class TransactionModel(models.Model):
    idtran = models.IntegerField(primary_key=True)
    sum = models.IntegerField()
    type = models.CharField(max_length=50)
    user = models.ForeignKey('UserModel')
    bank = models.ForeignKey('BankModel')
    date = models.DateTimeField()

```

myfile.html

```

{% load static %}
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <link href='{% static "polls/css/bootstrap.css" %}' rel="stylesheet">
    <title>{% block title %}{% endblock %}</title>
</head>
<body>
    <nav class="navbar navbar-light" style="background-color: #e3f2fd;">

    <div class="container-fluid">
        <!-- Brand and toggle get grouped for better mobile display -->
        <div class="navbar-header">
            <a class="navbar-brand"><em>untitled</em></a>
        </div>

        <!-- Collect the nav links, forms, and other content for toggling -->
        <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
            <ul class="nav navbar-nav">
                <li><a href="/orders_view/">Myfile.html</a></li>
                {% block header %}{% endblock %}
            </ul>
        </div><!-- /.navbar-collapse -->
    </div><!-- /.container-fluid -->
</nav>

    <div>
        {% block body %}Переопределяется в наследниках{% endblock %}
    </div>

```

```
</body>
</html>
```

about.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Title</title>
</head>
<body>
  Это тестовая страница!
</body>
</html>
```

transactions.html

```
{% extends 'myfile.html' %}

{% block title %}
  Lab6
{% endblock %}

{% block head %}
  Транзакции
{% endblock %}

{% block body %}
  Список транзакций:
  <p>id Сумма Тип id юзера id банка
  {% for i in transactions %}
  <br> {{i.idtran}}  {{i.sum}}  {{i.type}}  {{i.user_id}}  {{i.bank_id}}

  {% empty %}
  Список пуст
  {% endfor %}
{% endblock %}
```

Результаты работы программы

