Московский государственный технический университет им. Н.Э.Баумана

Кафедра «Системы обработки информации и управления»

Утверждаю: _____ Согласовано: _____

"___"_____2016 г. "___"_____2016 г.

«Введение в python»

Отчет по лабораторной работе №6

(вид документа)

писчая бумага формата А4

(вид носителя)

_____

(количество листов)

Исполнитель: студент группы РТ5-51

_____ Попков В.Е.

"_____"_____ 2016 г.

Москва – 2016

Задания лабораторной:
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

1)

```
from mysql import connector
connection = connector.connect(user='root', password='root', host='127.0.0.1', database='1db')
def save(id, name, age):
    id = id
    name = name
    age = age
    c = connection.cursor()
    c.execute("INSERT INTO users (id, name, age) VALUES (%s,%s,%s)",(id, name, age))
    connection.commit()
    c.close()

try:
    c = connection.cursor()
    c.execute("SELECT * from users;")
    take = c.fetchall()
    for e in take:
        print(e)
    c.execute("SELECT name from users WHERE age>=20;")
    take = c.fetchall()
    for e in take:
        print(e[0])
    pass
finally:
    connection.close()
```

2)

```
from mysql import connector
class conn():
    def __init__(self):
        self.connection = connector.connect(user='root', password='root', host='127.0.0.1', database='1db')
    def spisok_id(self):
        c = self.connection.cursor()
        c.execute("SELECT id from users;")
        return (c.fetchall())
    def spisok_all(self):
        c = self.connection.cursor()
        c.execute("SELECT * from users;")
        return (c.fetchall())
    def age(self,id):
        self.id=int(id)
        c = self.connection.cursor()
        c.execute("SELECT name,age from users WHERE id = %s"% self.id)
        return (c.fetchall())

class add:
    def __init__(self):
        self.connection = connector.connect(user='root', password='root', host='127.0.0.1', database='1db')

    def save(self,id,name,age):
        self.id = id
        self.name = name
        self.age = age
        c = self.connection.cursor()
        c.execute("INSERT INTO users (id, name, age) VALUES (%s,%s,%s)", (self.id,self.name,self.age))
        self.connection.commit()
        c.close()
```

PEP 8: indentation contains tabs

Platform and Plugin Updates
PyCharm is ready to update.

3)

```
from django.db import models
from django.utils import timezone


class Post(models.Model):
    author = models.ForeignKey('auth.User')
    title = models.CharField(max_length=200)
    text = models.TextField()
    created_date = models.DateTimeField(
            default=timezone.now)
    published_date = models.DateTimeField(
            blank=True, null=True)

    def publish(self):
        self.published_date = timezone.now()
        self.save()

    def __str__(self):
        return self.title# Create your models here.

class Man(models.Model):
    id = models.IntegerField(primary_key=True)
    name = models.CharField(max_length=200)
    age = models.IntegerField()
```

4)

```
content=[...]
from mysql import connector
class conn():
    def __init__(self):
        self.connection = connector.connect(user='root', password='root', host='127.0.0.1', database='1db')
    def spisok_id(self):
        c = self.connection.cursor()
        c.execute("SELECT id from users;")
        return (c.fetchall())
    def spisok_all(self):
        c = self.connection.cursor()
        c.execute("SELECT * from users;")
        return (c.fetchall())
    def age(self,id):
        self.id=int(id)
        c = self.connection.cursor()
        c.execute("SELECT name,age from users WHERE id = %s"% self.id)
        return (c.fetchall())
class add:...
def books(request):...
def govorit(request, pk):...
class MyView(View):
    def get(self, request):
        us = conn()
        ta = us.spisok_all()
        take = []
        for i in ta:
            take.append(i)
        return render_to_response('blog/test.html', {'current': take})
```

База данных в MySQL