Московский государственный технический университет им. Н.Э. Баумана

Факультет «Информатика и системы управления» Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Разработка интернет приложений»

Отчет по рубежному контролю №2 Вариант В-21

Выполнил: студент группы ИУ5-53Б Сергеев М.Ю. Проверил: преподаватель каф. ИУ5 Гапанюк Ю.Е.

Описание предметной области:

Существуют две сущности: Оператор и Язык программирования.

Оператор имеет следующие атрибуты: номер (id), имя (name), частота употребления (frequency) и номер связанного языка программирования одинко-многим (lan_id).

Язык программирования имеет следующие атрибуты: номер (id), название (name) и используемость (usability). Задание сущностей и список их значений в Python:

Задание моделей Django ORM

```
class.Operator(models.Model):
    id = models.AutoField(primary_key=True)
    name = models.CharField(max_length=30, blank=True, null=True)
    frequency = models.IntegerField(blank=True, null=True)
    lan_id = models.ForeignKey('Language', models.DO_NOTHING, db_column='lan_id', blank=True, null=True)

class Meta:
    managed = False
    db_table = 'operator'

class_Language(models.Model):
    id = models.AutoField(primary_key=True)
    name = models.CharField(max_length=30, blank=True, null=True)

    vsability = models.IntegerField(blank=True, null=True)

class_Meta:
    managed = False
    db_table = 'language'
```

Для возможности добавления и удаления данных был использован rest framework.

Сериализаторы:

```
ifrom rest_framework import serializers
import RK2.models as models

iclass LanguageSerializer(serializers.ModelSerializer):
    class Meta:
        model = models.Language
        fields = ["id", "name"]

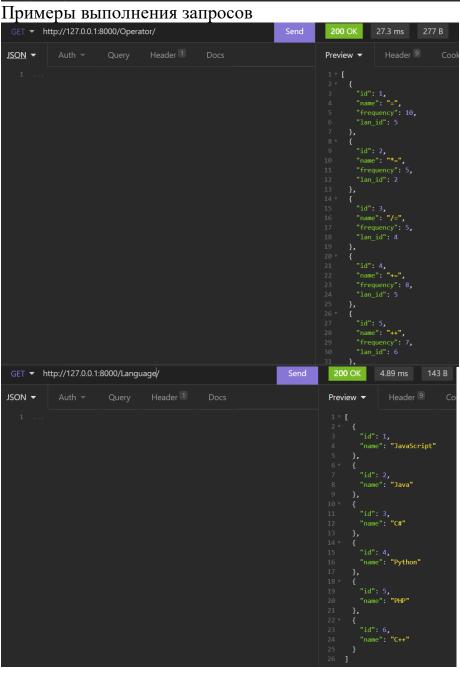
iclass OperatorSerializer(serializers.ModelSerializer):
    class Meta:
        model = models.Operator
        fields = ["id", "name", "frequency", "lan_id"]
```

Представления:

```
class LanguageViewSet(viewsets.ModelViewSet):
    queryset = models.Language.objects.all()
    serializer_class = serializers.LanguageSerializer

class OperatorViewSet(viewsets.ModelViewSet):
    queryset = models.Operator.objects.all()
    serializer_class = serializers.OperatorSerializer

def getOperators(Language: models.Language):
    result = []
    for op in models.Operator.objects.all():
        if op.lan_id == Language:
            result.append(op)
```



```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <title>Master</title>
   <style>
       table.collapse{
           border: 2px solid black;
           border-collapse: collapse;
          margin-top: 10px;
       td.upper{
           border: 2px solid black;
           padding: 10px;
    td{
       border: 2px solid black;
       padding: 5px;
  </style>
</head>
<body>
{% for value in values %}
  ID языка
       Название языка
    {{ value.0.id }}
      {{ value.0.name }}
    ID оператора
      Hазвание
      Частота использования
    {% for op in value.1 %}
      {{ op.id }}
        {{ op.frequency }}
```

```
{% endfor %}

{% endfor %}
</body>
</html>
```

| , v, ii ciii cz | | |
|-----------------|----------|-----------------------|
| ID языка | | Название языка |
| 1 | | JavaScript |
| ID оператора | Название | Частота использования |
| | | |
| ID языка | | Название языка |
| 2 | | Java |
| ID оператора | Название | Частота использования |
| 2 | *= | 5.4 |
| | | |
| ID языка | | Название языка |
| 3 | | C# |

| ID языка | | Название языка |
|--------------|----------|-----------------------|
| 3 | | C# |
| ID оператора | Название | Частота использования |
| 6 | | 3.6 |

| ID языка | | Название языка |
|--------------|----------|-----------------------|
| 4 | | Python |
| ID оператора | Название | Частота использования |
| 3 | /= | 5.1 |