МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

«ГОМЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ИМЕНИ П. О. СУХОГО»

Факультет автоматизированных и информационных систем

Кафедра «Информационные технологии»

дисциплина «Средства и технологии разработки информационных систем»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ

«Модели. Реляционные связи»

ВАРИАНТ № 22

Выполнил:

студент группы ИТП-31, Пронуза М.Ю.

Принял:

преподаватель Карась О.В.

Гомель 2024

**1. Цель работы:** освоить механизм организации связей между моделями.

**2. Ход работы и результаты.**

Задание: Доработать веб приложение, выполненное в предыдущей лабораторной работе. Необходимо, согласно предметной области, добавить еще одну модель и связь типа «многие ко одному». Реализовать параметризированный запрос, который выводит список всех сущностей на стороне отношений «многие» по запрошенной в параметре сущности на стороне «один».

Задание варианта 22 представлено на рисунке 1.



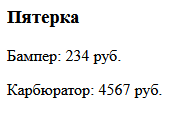
Рисунок 1 - вариант задания

Страница выбора машин представлена на рисунке 2.



Рисунок 2 - выбор машин

Страница выбора деталей для пятерки представлена на рисунке 3.

Рисунок 3 - выбор деталей

Код *djangoapp/urls.py* представлен в листинге A.

Код *djangoapp/settings.py* представлен в листинге Б.

Код *magazine/urls.py* представлен в листинге В.

Код *magazine/views.py* представлен в листинге Г.

Код *templates/carlist.html* представлен в листинге Д.

Код *magazine/models.py* представлен в листинге Е.

Код *magazine/sparelist.html* представлен в листинге Ё.

**3. Выводы:** были освоены механизмы организации связей между моделями.

**Листинг А**

from django.contrib import admin

from django.urls import include, path

urlpatterns = [

path('admin/', admin.site.urls),

path('', include("magazine.urls"))

]

**Листинг Б**

STATIC\_URL = 'static/'

STATICFILES\_DIRS = [

os.path.join(BASE\_DIR, "static"),

]

TEMPLATES = [

{

'BACKEND': 'django.template.backends.django.DjangoTemplates',

'DIRS': ['templates'],

'APP\_DIRS': True,

'OPTIONS': {

'context\_processors': [

'django.template.context\_processors.debug',

'django.template.context\_processors.request',

'django.contrib.auth.context\_processors.auth',

'django.contrib.messages.context\_processors.messages',

],

},

},

]

INSTALLED\_APPS = [

'django.contrib.admin',

'django.contrib.auth',

'django.contrib.contenttypes',

'django.contrib.sessions',

'django.contrib.messages',

'django.contrib.staticfiles',

'magazine'

]

**Листинг В**

from django.urls import path

from . import views

urlpatterns = [

path('/car/<int:car\_id>', views.sparelist, name='sparelist'),

path("", views.carlist, name="carlist"),

]

**Листинг Г**

from django.http import HttpResponse

from django.shortcuts import render

from magazine.models import Spare, Car

def sparelist(request, car\_id):

car = Car.objects.get(pk=car\_id)

spare\_list = Spare.objects.filter(car=car)

return render(request, 'sparelist.html', {'car\_name': car.name,'spare\_list': spare\_list})

def carlist(request):

car\_list = Car.objects.all()

return render(request, 'carlist.html', {'car\_list': car\_list})

**Листинг Д**

<!doctype html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport"

content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<title>List car</title>

</head>

<body>

<ul>

{% for i in car\_list %}

<li><a href="{% url 'sparelist' car\_id=i.id %}">{{i.name}}</a></li>

{% endfor %}

</ul>

</body>

</html>

**Листинг Е**

from django.db import models

class Car(models.Model):

name = models.CharField('название', max\_length=100, unique=True)

def \_\_str\_\_(self):

return str(self.name)

class Spare(models.Model):

car = models.ForeignKey(Car, on\_delete=models.CASCADE)

name = models.CharField('название', max\_length=100)

price = models.IntegerField('цена')

def \_\_str\_\_(self):

return str(self.name)

**Листинг Ё**

<!doctype html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport"

content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<title>{{car\_name}}</title>

</head>

<body>

<h3>{{car\_name}}</h3>

{% for i in spare\_list %}

<p data-car-name="{{i.car.name}}">{{i.name}}: {{i.price}} руб.</p>

{% endfor %}

</body>

</html>