

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
ИТМО**

Дисциплина: Основы веб-программирования

Отчет по лабораторной работе №1
Вариант 4

Создание сайта на Django

Выполнила:
Ларионова Мария
Группа: К3343

Проверил:
Говоров А. И.

2020 г.

Выполнение работы:

Модель данных:



Главная таблица это, разумеется, Tour. У одного агентства может быть много туров, а у одного тура, в свою очередь, может быть несколько условий для оплаты.

Реализация:

models.py:

```

# Create your models here.
class Agency(models.Model):
    name = models.CharField(max_length=50)

    def __str__(self):
        return "{}".format(self.name)

class PaymentCondition(models.Model):
    CONDITIONS_LIST = [
        ('1', '100% before the trip'),
        ('2', '50% before the trip, 50% after'),
        ('3', '70% before the trip, 30% after'),
        ('4', '10% off'),
        ('5', '90% before the trip, 10% after'),
        ('6', '60% before the trip, 40% after')
    ]

    conditions = models.CharField(choices=CONDITIONS_LIST, default='1', max_length=1)

    def __str__(self):
        return "Payment conditions: {}".format(self.get_conditions_display())

class Tour(models.Model):
    name = models.CharField(max_length=50)
    desc = models.CharField(max_length=1000)
    start_date = models.DateField()
    end_date = models.DateField()

    payment_conditions = models.ForeignKey(PaymentCondition, on_delete=models.CASCADE)
    agency = models.ForeignKey(Agency, on_delete=models.CASCADE)

    def __str__(self):
        return "{}: {}, {} - {}".format(self.agency, self.name, self.start_date, self.end_date)

class TourComment(models.Model):
    COMMENT_TYPES = [
        ('1', 'Tour content question'),
        ('2', 'Payment question'),
        ('3', 'Review')
    ]

    tour = models.ForeignKey(Tour, on_delete=models.CASCADE)

    comment_text = models.CharField(max_length=1000)
    question_type = models.CharField(choices=COMMENT_TYPES, default='1', max_length=1)
    author = models.ForeignKey(User, on_delete=models.CASCADE)

```

Сразу из моделей я стараюсь возвращать всю основную информацию, чтобы минимально делать это в шаблонах и сразу быстро ориентироваться в базе данных.

forms.py:

```

from django import forms
from .models import TourComment
from django.contrib.auth.models import User

class CommentForm(forms.ModelForm):
    class Meta:
        model = TourComment
        fields = [
            "question_type",
            "comment_text"
        ]

class UserForm(forms.ModelForm):
    password = forms.CharField(widget=forms.PasswordInput())
    class Meta():
        model = User
        fields = ('email', 'first_name', 'last_name', 'username', 'password')

```

views.py:

```

from django.shortcuts import render
from django.http import Http404, HttpResponseRedirect, HttpResponse
from django.views.generic.list import ListView
from django.contrib.auth import authenticate, login
from django.contrib.auth.decorators import login_required
from .models import *
from .forms import *

# Create your views here.
class TourView(ListView):
    model = Tour

    def get(self, request):
        context = {}

        try:
            context["tours"] = Tour.objects.all()
        except Tour.DoesNotExist:
            raise Http404("Tour does not exist")

        return render(request, 'tours.html', context)

```

```

@login_required
def about_tour(request, tour_id):

    try:
        tour = Tour.objects.get(pk=tour_id)

    except Tour.DoesNotExist:
        raise Http404("Tour does not exist")

    try:
        comments = TourComment.objects.filter(tour=tour_id).all()

    except TourComment.DoesNotExist:
        print("There are no comments")

    if request.POST.get('user_id'):
        user_id = int(request.POST.get('user_id'))

    form = CommentForm(request.POST)

    if form.is_valid():

        tour_form = form.save(commit=False)

        tour_form.post = form
        tour_form.author_id = user_id
        tour_form.tour_id = tour_id

        tour_form.save()

        return HttpResponseRedirect('/tours/{}/'.format(tour_id))

    return render(request, 'tour.html', {'tour': tour, 'comments': comments, 'form': form})

```

Запретила доступ к странице тура без авторизации.

```

def register(request):
    registered = False

    if request.method == 'POST':

        register_user_form = UserForm(data=request.POST)

        if register_user_form.is_valid():

            user = register_user_form.save()
            user.set_password(user.password)
            user.save()
            registered = True

        else:

            print(register_user_form.errors)

    else:

        register_user_form = UserForm()

    return render(request, 'reg.html', {'register_user_form': register_user_form, 'registered': registered})

def auth(request):
    if request.method == 'POST':

        username = request.POST.get('username')
        password = request.POST.get('password')

        user = authenticate(username=username, password=password)

        if user:

            if user.is_active:
                login(request, user)
                return HttpResponseRedirect('/tours/')

            else:
                return HttpResponse("Your account was inactive.")

        else:
            return HttpResponse("Incorrect pair login/password")

    else:
        return render(request, 'auth.html', {})

```

Стандартные views для регистрации и авторизации.

urls.py:

```

from django.contrib import admin
from django.urls import path, include
from .views import *

urlpatterns = [
    path('tours/', TourView.as_view()),
    path('tours/<int:tour_id>', about_tour),
    path('accounts/login/', auth),
    path('accounts/register/', register)
]

```

Страница со всеми турами:

Tours:

Global Travel: Scandinavian weekends, 2019-12-02 - 2019-12-18

[Show more information](#)

Global Travel: New Zealand lifestyle, 2020-02-11 - 2020-02-25

[Show more information](#)

Global Travel: European tour, 2020-02-29 - 2020-03-10

[Show more information](#)

Latin America Travel: Vamos vamos, 2020-04-18 - 2020-04-27

[Show more information](#)

Latin America Travel: Maya secrets, 2020-04-18 - 2020-04-25

[Show more information](#)

Latin America Travel: Patagonia Beauty, 2020-04-29 - 2020-05-18

[Show more information](#)

Exotic Travel: Exploring India, 2021-04-18 - 2021-05-18

[Show more information](#)

Exotic Travel: African Safari, 2020-03-18 - 2020-03-25

[Show more information](#)

Exotic Travel: Taste of Asia, 2019-04-18 - 2019-04-23

[Show more information](#)

До двоеточия идёт название турфирмы, а после название самого тура, а завершает это всё даты начала и конца.

Tour:

Global Travel: Scandinavian weekends, 2019-12-02 - 2019-12-18

Tour description:

Spend a holiday in Sweden, Finland and Norway and learn how locals live

Question type: Tour content question

What cities will we go to?

Authored by: maria

Add new comment:

- This field is required.

Question type: Tour content question ▾

- This field is required.

Comment text:

Add comment

Так выглядит страница, где выводится более подробная информация по туру и комментарии к нему.

Sorry! You should [Register](#) or [Login](#) before adding your own comment.

Так будет выглядеть страница без авторизации.

Страница регистрации:

Email address:

First name:

Last name:

Username: Required. 150 characters or fewer. Letters, digits and @/./+/-/_ only.

Password:

После регистрации мы получаем такое сообщение на экране:

[Login](#) for getting full access for this site.

Страница для авторизации:

Username:

Password:

Don't have account yet? [Register now](#)

Добавила комментарий от нового пользователя:

Tour:

Global Travel: Scandinavian weekends, 2019-12-02 - 2019-12-18

Tour description:

Spend a holiday in Sweden, Finland and Norway and learn how locals live

Question type: Tour content question

What cities will we go to?

Authored by: marla

Question type: Payment question

How many it'll costs for 2 persons?

Authored by: JohnDoe

Add new comment:

- This field is required.

Question type:

- This field is required.

Comment text: