Московский государственный технический университет им. Н.Э. Баумана

Факультет «Информатика и системы управления»

Кафедра «Автоматизированные системы обработки информации и управления»



**Отчет**

**Домашнее задание**

**По курсу «Разработка интернет приложений»**

**ИСПОЛНИТЕЛЬ:**

Группа ИУ5-55Б

Нагдимаев М.А.

"13" января 2020 г.

**ПРЕПОДАВАТЕЛЬ:**

Гапанюк Ю.Е.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

"\_\_"\_\_\_\_\_\_\_\_\_\_\_2020 г.

Москва 2020

1. **Задание**

Создайте прототип веб-приложения с использованием фреймворка Django на основе базы данных, реализующий концепцию master/detail. Прототип должен содержать:

* Две модели, связанные отношением один-ко-многим.
* Стандартное средство администрирования Django позволяет редактировать данные моделей. Желательно настроить русификацию ввода и редактирования данных.
* Веб-приложение формирует отчет в виде отдельного view/template, отчет выводит HTML-страницу, содержащую связанные данные из двух моделей.
* Для верстки шаблонов используется фреймворк Bootstrap, или аналогичный фрейворк по желанию студента.

Доп. задание:

Реализация связи много-ко-многим (с возможностью редактирования данных в пользовательском интерфейсе) и развертывание приложения на облачном сервисе (например, heroku).

1. **Текст программы**

mysite\urls.py

from django.contrib import admin  
from django.urls import include, path  
  
urlpatterns = [  
 path('jutsu/', include('jutsu.urls')),  
 path('admin/', admin.site.urls),  
]

jutsu\urls.py

from django.urls import path  
from . import views  
  
app\_name = 'jutsu'  
urlpatterns = [  
 path('', views.index, name='index'),  
 path('<int:anime\_id>/', views.character, name='detail\_char'),  
 path('<str:genre\_name>', views.genres, name='index')  
]

jutsu \admin.py

from django.contrib import admin  
from .models import Anime, Сharacter, Anime\_Genre, Genre  
  
admin.site.register(Anime)  
admin.site.register(Сharacter)  
admin.site.register(Anime\_Genre)  
admin.site.register(Genre)

mysite\setting.py

*только часть, где добавляем наше приложение*

INSTALLED\_APPS = [  
 'django.contrib.admin',  
 'django.contrib.auth',  
 'django.contrib.contenttypes',  
 'django.contrib.sessions',  
 'django.contrib.messages',  
 'django.contrib.staticfiles',  
 'jutsu'  
]

jutsu\apps.py

from django.apps import AppConfig  
  
  
class JutsuConfig(AppConfig):  
 name = 'jutsu'

*для heroku*

DEBUG = False  
  
ALLOWED\_HOSTS = ['127.0.0.1', 'limitless-springs-90553.herokuapp.com']

STATIC\_ROOT = os.path.join(BASE\_DIR, 'staticfiles')

jutsu\models.py

*реализована связь многие ко многим*

from django.db import models  
  
  
class Genre(models.Model):  
 name = models.CharField(max\_length=50)  
  
 def \_\_str\_\_(self):  
 return self.name  
  
  
class Anime(models.Model):  
 name = models.CharField(max\_length=50)  
 release\_time = models.CharField(max\_length=10)  
 description = models.CharField(max\_length=700)  
 img = models.CharField(max\_length=200)  
 head\_img = models.CharField(max\_length=200)  
 genres = models.ManyToManyField(Genre, through='Anime\_Genre')  
  
  
 def \_\_str\_\_(self):  
 return self.name  
  
  
class Сharacter(models.Model):  
 name = models.ForeignKey(Anime, on\_delete=models.CASCADE)  
 name\_char = models.CharField(max\_length=50)  
 sag = models.CharField(max\_length=100)  
 side = models.CharField(max\_length=100)  
 img = models.CharField(max\_length=200)  
  
  
 def \_\_str\_\_(self):  
 return self.name\_char  
  
  
class Anime\_Genre(models.Model):  
 genre = models.ForeignKey(Genre, on\_delete = models.CASCADE)  
 anime = models.ForeignKey(Anime, on\_delete=models.CASCADE)

jutsu\views.py

from django.shortcuts import get\_list\_or\_404, render  
from django.http import HttpResponseRedirect  
from django.template import loader  
from .models import Anime, Сharacter, Genre, Anime\_Genre  
  
  
def index(request):  
 anime\_list = Anime.objects.all()  
 genre\_anime\_list = {}  
 for i in anime\_list:  
 anime\_genre\_list = get\_list\_or\_404(Anime\_Genre, anime=i.id)  
 genre\_list\_lol = []  
 for j in anime\_genre\_list:  
 genre\_list\_lol.append(j.genre.name)  
 genre\_anime\_list[i.id] = genre\_list\_lol  
 genre\_list = Genre.objects.all()  
 context = {'anime\_list': anime\_list, 'genre\_anime\_list': genre\_anime\_list, 'genre\_list': genre\_list}  
 return render(request, 'jutsu/index.html', context)  
  
   
def character(request, anime\_id):  
 anime\_list = Anime.objects.all()  
 anime = 0  
 for i in anime\_list:  
 if i.id == anime\_id:  
 anime = i  
 if request.method == "GET":  
 char\_list = get\_list\_or\_404(Сharacter, name=anime\_id)  
 return render(request, 'jutsu/detail\_char.html', {'anime': anime,  
 'char\_list': char\_list})  
 elif request.method == "POST":  
 anime.name = request.POST.get("anime\_name")  
 anime.release\_time = request.POST.get("anime\_release")  
 anime.save()  
 return HttpResponseRedirect(f"/jutsu/{anime\_id}/")  
  
  
def genres(request, genre\_name):  
 genre\_list = Genre.objects.all()  
 for i in genre\_list:  
 if i.name == genre\_name:  
 genre = i.id  
 anime\_genre\_list = get\_list\_or\_404(Anime\_Genre, genre=genre)  
 anime\_list = []  
 for i in anime\_genre\_list:  
 anime\_list.append(i.anime)  
 return render(request, 'jutsu/index.html', {'genre\_name': genre\_name,  
 'anime\_list': anime\_list,  
 'genre\_list': genre\_list})

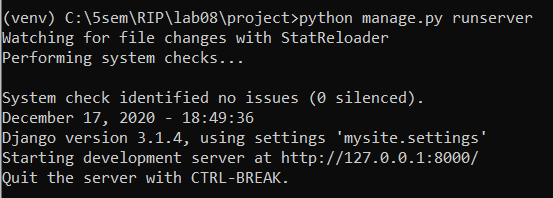
jutsu\templates\jutsu\index.html

<!doctype html>  
<html lang="ru">  
<head>  
 <!-- Обязательные метатеги -->  
 <meta charset="utf-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
  
 <!-- Bootstrap CSS -->  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-giJF6kkoqNQ00vy+HMDP7azOuL0xtbfIcaT9wjKHr8RbDVddVHyTfAAsrekwKmP1" crossorigin="anonymous">  
 <script src="https://kit.fontawesome.com/941547f2e9.js" crossorigin="anonymous"></script>  
 <style>  
 html {  
 overflow-x: hidden;  
 overflow-y: scroll;  
 }  
 main {  
 overflow-x: hidden;  
 }  
 footer {  
 overflow-x: hidden;  
 }  
 .signup>input:nth-of-type(1):invalid~input:nth-of-type(2):invalid~.progress>.progress-bar {  
 width: 0%;  
 }  
 .signup>input:nth-of-type(1):valid~input:nth-of-type(2):invalid~.progress>.progress-bar {  
 width: 50%;  
 }  
 .signup>input:nth-of-type(1):invalid~input:nth-of-type(2):valid~.progress>.progress-bar {  
 width: 50%;  
 }  
 .signup>input:nth-of-type(1):valid~input:nth-of-type(2):valid~.progress>.progress-bar {  
 width: 100%;  
 }  
 .flex {  
 display: flex;  
 justify-content: space-between;  
 }  
 </style>  
  
 <title>HW</title>  
</head>  
<body>  
<main class="bg-dark">  
 <nav class="navbar navbar-expand-lg navbar-dark bg-dark">  
 <div class="container-fluid modal-header">  
 <a class="navbar-brand" href="{% url 'jutsu:index' %}">JUTSU</a>  
 <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">  
 <span class="navbar-toggler-icon"></span>  
 </button>  
 <div class="collapse navbar-collapse" id="navbarSupportedContent">  
 <ul class="navbar-nav mr-auto mb-2 mb-lg-0">  
 <li class="nav-item dropdown">  
 <a class="nav-link dropdown-toggle active" href="#" id="navbarDropdown" role="button" data-toggle="dropdown" aria-expanded="false">  
 Жанры  
 </a>  
 <ul class="dropdown-menu" aria-labelledby="navbarDropdown">  
 {% if genre\_list %}  
 {% for genre in genre\_list %}  
 <li><a class="dropdown-item" href="{% url 'jutsu:index' genre.name %}">{{genre.name}}</a></li>  
 {% endfor %}  
 {% endif %}  
 </ul>  
 </li>  
 </ul>  
 </div>  
 {% if genre\_name %}  
 <div class="navbar-brand">  
 <h2>{{genre\_name}}</h2>  
 </div>  
 <a class="navbar-brand" href="{% url 'jutsu:index' %}">Все жанры</a>  
 {% endif %}  
 </div>  
 </nav>  
  
 {% if anime\_list %}  
 {% for anime in anime\_list %}  
 <table class="table table-dark table-borderless">  
 <tbody>  
 <tr>  
 <td class="align-bottom" style="text-align: right" width="60%">  
 <a href="{% url 'jutsu:detail\_char' anime.id %}"><h2>{{anime.name}}</h2></a>  
 </td>  
 <td class="align-middle" rowspan="4">  
 <a href="{% url 'jutsu:detail\_char' anime.id %}"><img src={{anime.img}} class="rounded img-fluid" style="max-width: 90%; height: auto"></a>  
 </td>  
 </tr>  
 <tr>  
 <td class="align-bottom" style="text-align: right" width="60%">  
 <h5>Жанры: </h5>  
 {% for genre in anime.genres.all %}  
 <a href="{% url 'jutsu:index' genre %}">  
 <h5>{{genre}}</h5>  
 </a>  
 {% endfor %}  
 </td>  
 </tr>  
 <tr>  
 <td class="align-bottom" style="text-align: right" width="60%">  
 <h3>Дата выхода: {{anime.release\_time}}</h3>  
 </tr>  
 <tr>  
 <td class="align-middle" style="text-align: right" width="60%">{{anime.description}}  
 </tr>  
 </tbody>  
 </table>  
 {% endfor %}  
 {% else %}  
 <p>No group are available.</p>  
 {% endif %}  
</main>  
<footer class="bg-dark modal-footer">  
 <address class="container-fluid">  
 <p style="color: #d3d3d4">Контакты: +7-(989)-452-81-03</p>  
 </address>  
 <p class="container-fluid text-center" style="color: #d3d3d4">2020 BMSTU</p>  
</footer>  
  
 <script>  
 // JavaScript for disabling form submissions if there are invalid fields  
 (function () {  
 'use strict'  
  
 // Fetch all the forms we want to apply custom Bootstrap validation styles to  
 var forms = document.querySelectorAll('.needs-validation')  
  
 // Loop over them and prevent submission  
 Array.prototype.slice.call(forms)  
 .forEach(function (form) {  
 form.addEventListener('submit', function (event) {  
 if (!form.checkValidity()) {  
 event.preventDefault()  
 event.stopPropagation()  
 }  
  
 form.classList.add('was-validated')  
 }, false)  
 })  
 })()  
 </script>  
<!-- Optional JavaScript -->  
<!-- Popper.js first, then Bootstrap JS -->  
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js" integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>  
<script src="https://stackpath.bootstrapcdn.com/bootstrap/5.0.0-alpha1/js/bootstrap.min.js" integrity="sha384-oesi62hOLfzrys4LxRF63OJCXdXDipiYWBnvTl9Y9/TRlw5xlKIEHpNyvvDShgf/" crossorigin="anonymous"></script>  
</body>  
</html>

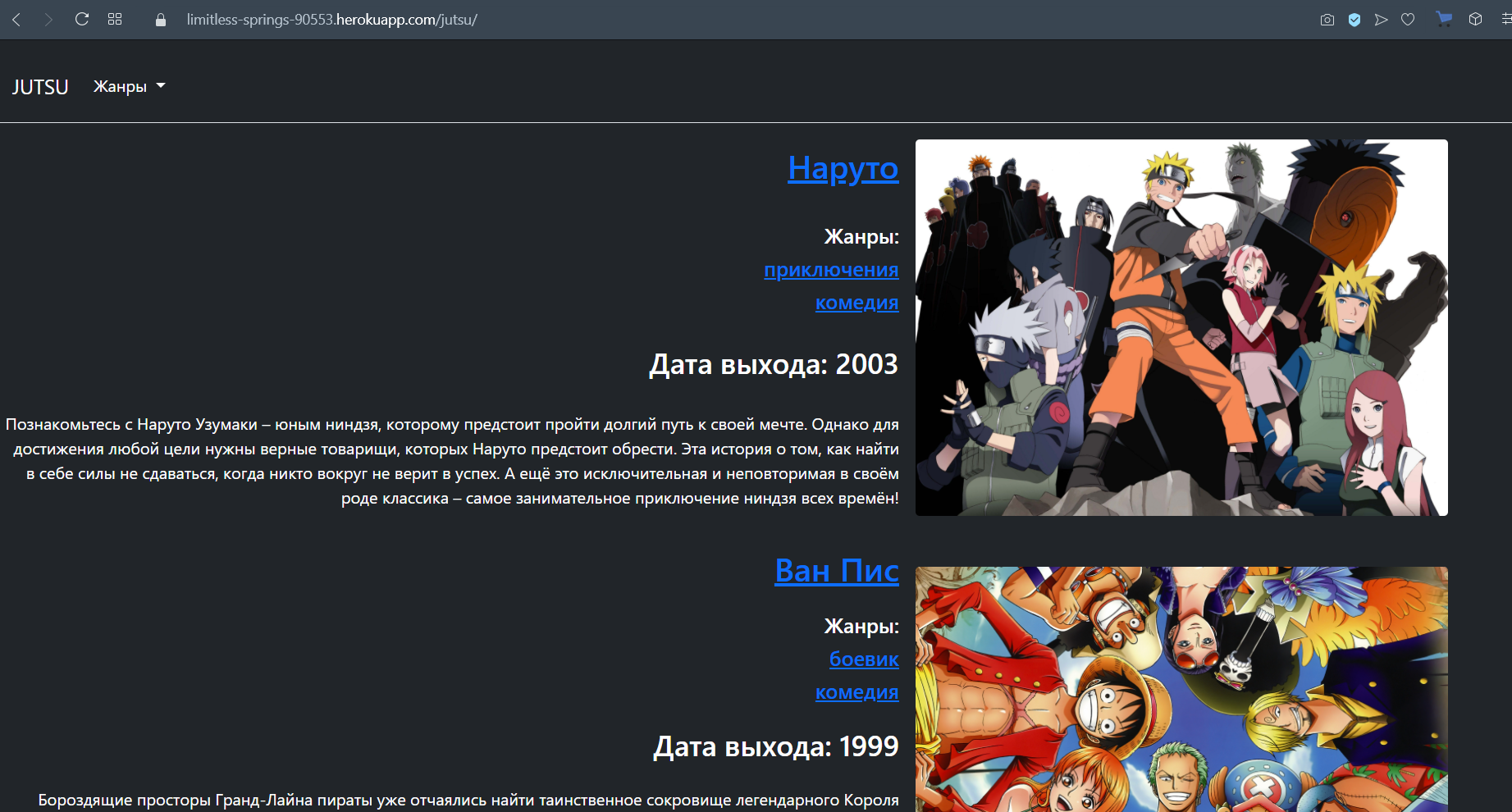
jutsu\templates\jutsu\detail\_char.html

<!doctype html>  
<html lang="ru">  
<head>  
 <!-- Обязательные метатеги -->  
 <meta charset="utf-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 {% load static %}  
 <!-- Bootstrap CSS -->  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta1/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-giJF6kkoqNQ00vy+HMDP7azOuL0xtbfIcaT9wjKHr8RbDVddVHyTfAAsrekwKmP1" crossorigin="anonymous">  
 <script src="https://kit.fontawesome.com/941547f2e9.js" crossorigin="anonymous"></script>  
 <style>  
 html {  
 overflow-x: hidden;  
 overflow-y: scroll;  
 }  
 main {  
 overflow-x: hidden;  
 }  
 footer {  
 overflow-x: hidden;  
 }  
 .signup>input:nth-of-type(1):invalid~input:nth-of-type(2):invalid~.progress>.progress-bar {  
 width: 0%;  
 }  
 .signup>input:nth-of-type(1):valid~input:nth-of-type(2):invalid~.progress>.progress-bar {  
 width: 50%;  
 }  
 .signup>input:nth-of-type(1):invalid~input:nth-of-type(2):valid~.progress>.progress-bar {  
 width: 50%;  
 }  
 .signup>input:nth-of-type(1):valid~input:nth-of-type(2):valid~.progress>.progress-bar {  
 width: 100%;  
 }  
 </style>  
  
 <title>JUTSU</title>  
</head>  
<body>  
<main class="bg-dark">  
 <nav class="navbar navbar-expand-lg navbar-dark bg-dark">  
 <div class="container-fluid modal-header">  
 <a class="navbar-brand" href="{% url 'jutsu:index' %}">JUTSU</a>  
 <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">  
 <span class="navbar-toggler-icon"></span>  
 </button>  
 <div class="navbar-brand">  
 <h2>{{anime.name}}</h2>  
 </div>  
 <div class="float-end">  
 <!-- Button trigger modal -->  
 <button type="button" class="btn btn-primary green\_button" data-toggle="modal" data-target="#signUpModal">  
 Изменить  
 </button>  
  
 <!-- Modal -->  
 <div class="modal fade" id="signUpModal" tabindex="-1" aria-labelledby="signUpModalLabel" aria-hidden="true">  
 <div class="modal-dialog">  
 <div class="modal-content">  
 <div class="modal-header">  
 <h5 class="modal-title" id="signUpModalLabel">Внести изменения</h5>  
 <button type="button" class="close" data-dismiss="modal" aria-label="Close">  
 <span aria-hidden="true">&times;</span>  
 </button>  
 </div>  
 <div class="modal-body">  
 <form class="g-3 needs-validation signup" method="post" novalidate>  
 {% csrf\_token %}  
 <label for="validationCustom01" class="form-label">Название аниме</label>  
 <input type="text" class="form-control" id="validationCustom01" value={{anime.name}} name="anime\_name" required>  
 <div class="invalid-feedback">  
 Пожалуйста введите название аниме  
 </div>  
 <p></p>  
 <label for="validationCustom02" class="form-label">Дата выхода</label>  
 <input type="text" class="form-control" id="validationCustom02" value={{anime.release\_time}} name="anime\_release" required>  
 <div class="invalid-feedback">  
 Пожалуйста введите дату выхода  
 </div>  
 <p></p>  
 <div class="modal-footer">  
 <input class = "btn btn-primary green\_button" type="submit" value="Save" name="\_save">  
 <button class="btn btn-secondary" type="button" data-dismiss="modal">Close</button>  
 </div>  
 <script>  
 // JavaScript for disabling form submissions if there are invalid fields  
 (function () {  
 'use strict'  
  
 // Fetch all the forms we want to apply custom Bootstrap validation styles to  
 var forms = document.querySelectorAll('.needs-validation')  
  
 // Loop over them and prevent submission  
 Array.prototype.slice.call(forms)  
 .forEach(function (form) {  
 form.addEventListener('submit', function (event) {  
 if (!form.checkValidity()) {  
 event.preventDefault()  
 event.stopPropagation()  
 }  
  
 form.classList.add('was-validated')  
 }, false)  
 })  
 })()  
 </script>  
 </form>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 </nav>  
  
 {% if anime.id == 3 %}  
 <img src="{% static "jutsu/images/head/bibop.jpg" %}" alt="lol" class="rounded img-fluid">  
 {% else %}  
 <img src={{anime.head\_img}}>  
 {% endif %}  
 {% if char\_list %}  
 {% for char in char\_list %}  
 <div class="row">  
 <div class="col-8">  
 <table class="table table-dark table-borderless">  
 <tbody>  
 <tr>  
 <td class="align-bottom" style="text-align: right" width="40%" height="30">  
 <h1>{{char.name\_char}}</h1>  
 </td>  
 <td class="align-middle" rowspan="4">  
 <img src={{char.img}} class="rounded img-fluid" style="max-width: 100%; height: auto">  
 </td>  
 </tr>  
 <tr>  
 <td class="align-bottom" style="text-align: right" width="40%" height="30">  
 <h4>{{char.sag}}</h4>  
 </tr>  
 <tr>  
 <td class="align-middle" style="text-align: right" width="40%" height="30"><h4>{{char.side}}</h4>  
 </tr>  
 </tbody>  
 </table>  
 </div>  
 </div>  
 {% endfor %}  
 {% else %}  
 <p>No group are available.</p>  
 {% endif %}  
</main>  
<footer class="bg-dark modal-footer">  
 <address class="container-fluid">  
 <p style="color: #d3d3d4">Контакты: +7-(989)-452-81-03</p>  
 </address>  
 <p class="container-fluid text-center" style="color: #d3d3d4">2020 BMSTU</p>  
</footer>  
  
 <script>  
 // JavaScript for disabling form submissions if there are invalid fields  
 (function () {  
 'use strict'  
  
 // Fetch all the forms we want to apply custom Bootstrap validation styles to  
 var forms = document.querySelectorAll('.needs-validation')  
  
 // Loop over them and prevent submission  
 Array.prototype.slice.call(forms)  
 .forEach(function (form) {  
 form.addEventListener('submit', function (event) {  
 if (!form.checkValidity()) {  
 event.preventDefault()  
 event.stopPropagation()  
 }  
  
 form.classList.add('was-validated')  
 }, false)  
 })  
 })()  
 </script>  
<!-- Optional JavaScript -->  
<!-- Popper.js first, then Bootstrap JS -->  
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js" integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>  
<script src="https://stackpath.bootstrapcdn.com/bootstrap/5.0.0-alpha1/js/bootstrap.min.js" integrity="sha384-oesi62hOLfzrys4LxRF63OJCXdXDipiYWBnvTl9Y9/TRlw5xlKIEHpNyvvDShgf/" crossorigin="anonymous"></script>  
</body>  
</html>

**Экранные формы с примерами выполнения программы**



https://limitless-springs-90553.herokuapp.com/jutsu/





Форма для редактирования (доп задание)

