

Пример 1. Работа с формой

<http://127.0.0.1:8000/form/>

Данные:

Введите имя :

Введите возраст (полных лет) :

Языки программирования:

Результат отображения на странице

<http://127.0.0.1:8000/form/postuser/>

Имя пользователя : Анна

Возраст пользователя 20

Languages: ['C++', 'Java', 'C#', 'Python']

Программный код файла view.py

```
import numpy as np
import matplotlib.pyplot as plt
import skfuzzy as fuzz
from django.http import HttpResponse
from django.shortcuts import render

def form(request):
    return render(request, "form.html")

def postuser(request):
    # получаем из данных запроса POST отправленные через форму данные
    name = request.POST.get("name", "Undefined")
    age = request.POST.get("age", 7)
    langs = request.POST.getlist("languages", ["python"])
    #return HttpResponse(f"<h2> Имя пользователя {name} и Возраст пользователя {age}</h2>")

    return HttpResponse(f"""
        <div> Имя пользователя : {name} </div> <br/> <br/>
        <div> Возраст пользователя {age} </div> <br/> <br/>
        <div> Languages: {langs} </div> <br/> <br/>
        """)
```

Программный код файла Form.html

```
<!DOCTYPE html>
<html>
<html lang="ru">
<head>
    <meta charset="utf-8" />
    <title>FORM</title>
</head>
<body>
    <h2> Данные: </h2>
    <form method="post" action="postuser/">
        {% csrf_token %}
        <p> Введите имя :<br> <input name="name" /> </p>
        <p> Введите возраст (полных лет) :<br> <input name="age" type="number" /></p>
        <p>
            Языки программирования:<br />
            <p> <input name="languages" /><br /> </p>
            <p><input name="languages" /><br /> </p>
            <p> <input name="languages" /><br /> </p>
            <p> <input name="languages" /><br /> </p>
        </p>

        <input type="submit" value="Send" />
```

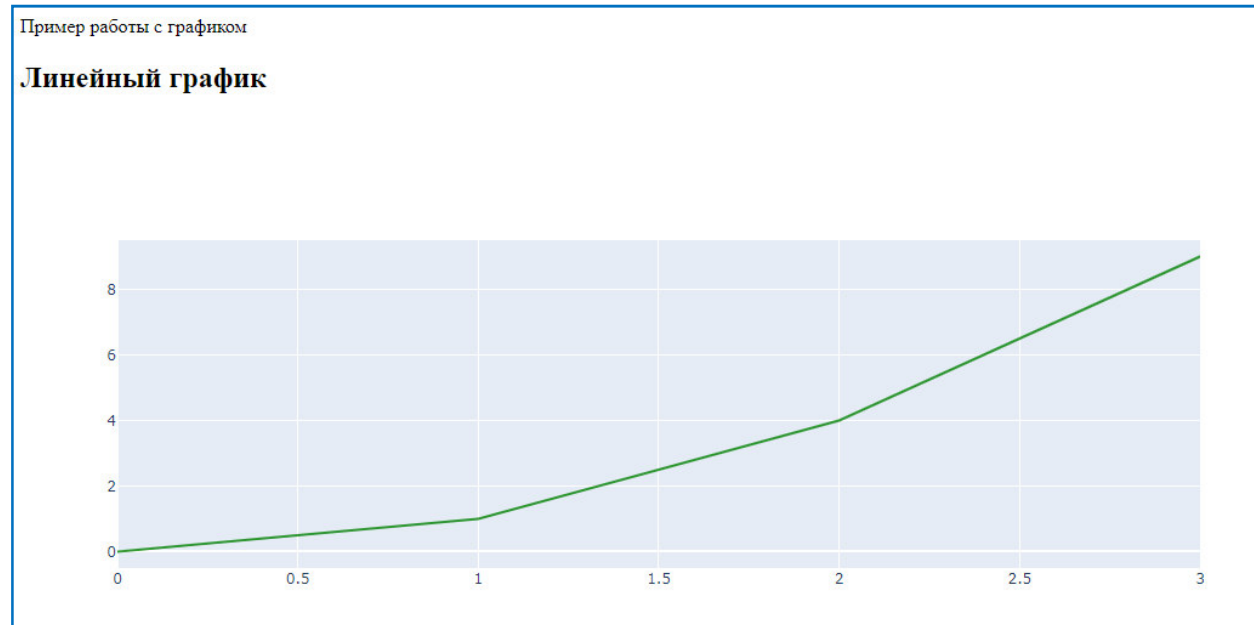
```
</form>  
</body>  
</html>
```

Программный код файла `urls.py`

```
from django.urls import path  
from hello import views  
  
urlpatterns = [  
    path("form/", views.form),  
    path("form/postuser/", views.postuser),  
]
```

Пример 2. Работа с графиком функции

<http://127.0.0.1:8000/graph/>



Программный код graph.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title><i> Пример работы с графиком </i></title>
</head>
<body>

  <h2>Линейный график </h2>
  {% autoescape off %}
  {{ plot_div }}
  {% endautoescape %}

</body>
</html>
```

Программный код views.py

```
import numpy as np
import matplotlib.pyplot as plt
import skfuzzy as fuzz
from django.http import HttpResponse

from django.shortcuts import render
from plotly.offline import plot
from plotly.graph_objs import Scatter

def graph(request):
    #return render(request, "graph.html")
    x_data = [0,1,2,3]
```

```

y_data = [x**2 for x in x_data]
plot_div = plot([Scatter(x=x_data, y=y_data,
                        mode='lines', name='test',
                        opacity=0.8, marker_color='green')],
               output_type='div')
return render(request, "graph.html", context={'plot_div': plot_div})

```

Программный код urls.py

```

from django.urls import path
from hello import views

urlpatterns = [
    #path("", views.index),
    #path("about/", views.about),
    #path("contact/", views.contact),
    path("graph/", views.graph),

    #path("form/", views.form),
    #path("form/postuser/", views.postuser),

    #path("selectform/", views.selectform),
    #path("selectform/postuser/", views.postuser),
]

```

Пример 3. Отображение данных на форме

Общие сведения о программе:

Нечеткие входные переменные системы управления:

Температура воздуха в помещении (входная переменная), градусы С

- Низкая {temp1_low_a, temp1_low_b, temp1_low_c}
- Средняя {temp1_md_a, temp1_md_b, temp1_md_c}
- Высокая {temp1_high_a, temp1_high_b, temp1_high_c}

Влажность воздуха (входная переменная), %

- Низкая {humidity_low_a, humidity_low_b, humidity_low_c}
- Средняя {humidity_md_a, humidity_md_b, humidity_md_c}
- Высокая {humidity_high_a, humidity_high_b, humidity_high_c}

Температура воздуха за окном (входная переменная) , градусы С

- Низкая {temp2_low_a, temp2_low_b, temp2_low_c}
- Средняя {temp2_md_a, temp2_md_b, temp2_md_c}
- Высокая {temp2_high_a, temp2_high_b, temp2_high_c}

Уровень отопления (выходная переменная), %

- Низкий {heating_low_a, heating_low_b, heating_low_c}
- Средний {heating_md_a, heating_md_b, heating_md_c}
- Высокий {heating_high_a, heating_high_b, heating_high_c}

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>About</title>
</head>
<body>
```

```
<h2> <i> Общие сведения о программе: </i> </h2>
```

```
  <h2><font size="3" color="black" face="Arial"> Нечеткие входные переменные системы
управления: </font> </h2>
```

```
    <p><b> <font size="3" color="blue" face="Arial"> Температура воздуха в
помещении (входная переменная), градусы С </font> </b></p>
```

```
    <ul>
```

```
    <li><p> <b>Низкая {temp1_low_a, temp1_low_b, temp1_low_c}</b></p></li>
```

```
    <li><p><b> Средняя {temp1_md_a, temp1_md_b, temp1_md_c} </b></p></li>
```

```
    <li><p><b> Высокая {temp1_high_a, temp1_high_b, temp1_high_c}</b></p></li>
```

```

        </ul>
        <p><b> <font size="3" color="green" face="Arial"> Влажность воздуха (входная
переменная), % </font> </b></p>
        <ul>
        <li><p> <b>Низкая {humidity_low_a, humidity_low_b,
humidity_low_c}</b></p></li>
        <li><p><b> Средняя {humidity_md_a, humidity_md_b, humidity_md_c}
</b></p></li>
        <li><p><b> Высокая {humidity_high_a, humidity_high_b,
humidity_high_c}</b></p></li>
        </ul>
        <p><b> <font size="3" color="DarkKhaki " face="Arial"> Температура воздуха за
окном (входная переменная) , градусы С </font> </b></p>
        <ul>
        <li><p> <b>Низкая {temp2_low_a, temp2_low_b, temp2_low_c}</b></p></li>
        <li><p><b> Средняя {temp2_md_a, temp2_md_b, temp2_md_c} </b></p></li>
        <li><p><b> Высокая {temp2_high_a, temp2_high_b, temp2_high_c}</b></p></li>
        </ul>

        <p><b> <font size="3" color="Maroon" face="Arial"> Уровень отопления
(выходная переменная), % </font> </b></p>
        <ul>
        <li><p> <b>Низкий {heating_low_a, heating_low_b, heating_low_c}</b></p></li>
        <li><p><b> Средний {heating_md_a, heating_md_b, heating_md_c} </b></p></li>
        <li><p><b> Высокий {heating_high_a, heating_high_b,
heating_high_c}</b></p></li>
        </ul>

</body>
</html>

```

Программный код файла view.py

```

from django.shortcuts import render
import numpy as np
import matplotlib.pyplot as plt
import skfuzzy as fuzz
from django.http import HttpResponse

from django.shortcuts import render
from plotly.offline import plot
from plotly.graph_objs import Scatter

def about1(request):
    return render(request, "about1.html")

```

Программный код urls.py

```
from django.urls import path
from hello import views

urlpatterns = [
    path("", views.index),
    path("about/", views.about),
    path("about1/", views.about1),
    path("contact/", views.contact),
    path("graph/", views.graph),

    path("form/", views.form),
    path("form/postuser/", views.postuser),

    path("selectform/", views.selectform),
    path("selectform/postuser/", views.postuser),
]
```

Пример 4. Работа с формами