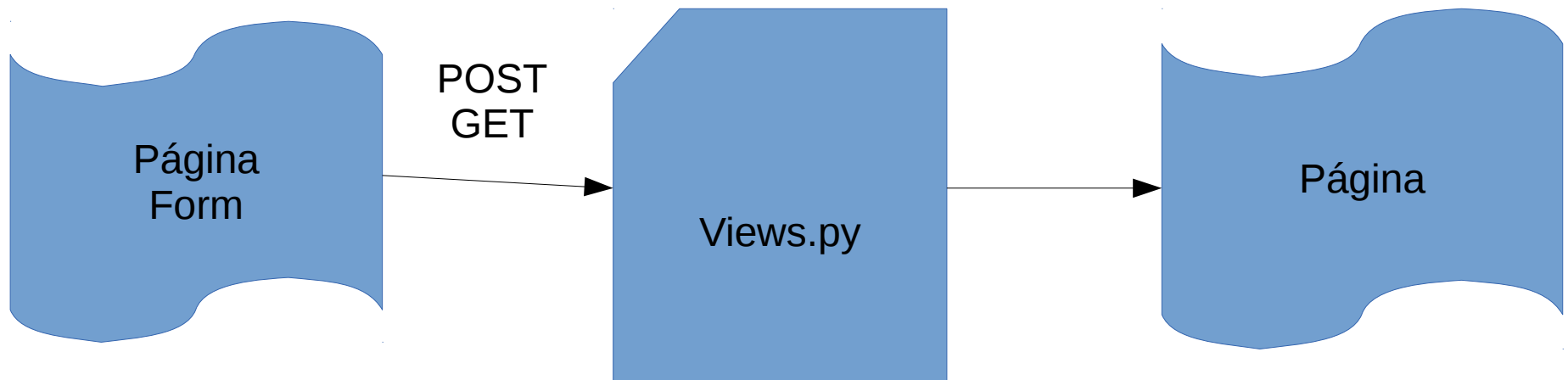


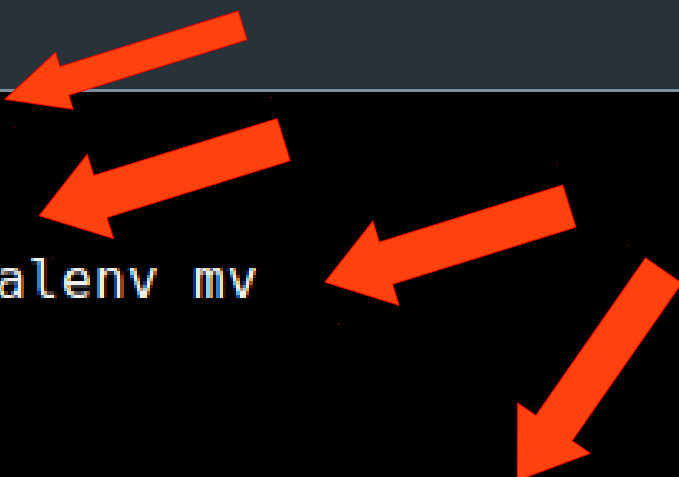
# Django Forms

# Um Form!



# Abra o Nitrous.IO ou em sua máquina, e crie um novo projeto!

```
× Console +  
action@unifran-78230:~$ mkdir projetoForm  
action@unifran-78230:~$ cd projetoForm  
action@unifran-78230:~/projetoForm$ virtualenv mv  
New python executable in mv/bin/python  
Installing setuptools, pip...done.  
action@unifran-78230:~/projetoForm$ source mv/bin/activate
```

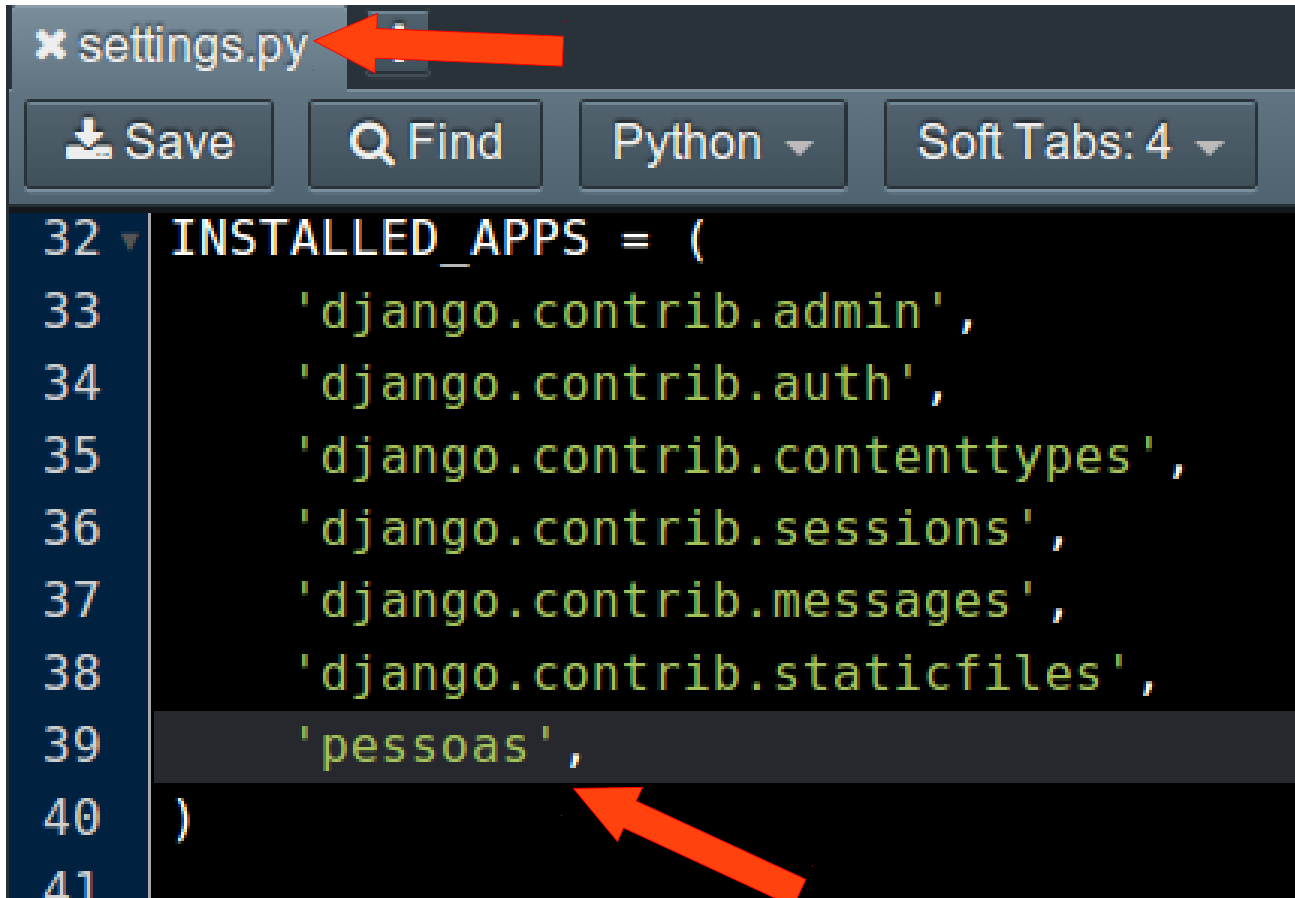


# Instale o Django 1.6 usando o PIP E crie os Objetos do sistema

```
(mv)action@unifran-78230:~/projetoForm$ pip install django
Downloading/unpacking django
  Downloading Django-1.6.5-py2.py3-none-any.whl (6.7MB): 6.7MB downloaded
Installing collected packages: django
Successfully installed django
Cleaning up...
(mv)action@unifran-78230:~/projetoForm$ django-admin.py startproject projeto
(mv)action@unifran-78230:~/projetoForm$ cd projeto/
(mv)action@unifran-78230:~/projetoForm/projeto$ python manage.py startapp pessoas
```

# Pronto!

# Codificando!



```
32 INSTALLED_APPS = (  
33     'django.contrib.admin',  
34     'django.contrib.auth',  
35     'django.contrib.contenttypes',  
36     'django.contrib.sessions',  
37     'django.contrib.messages',  
38     'django.contrib.staticfiles',  
39     'pessoas',  
40 )  
41
```

# Criando o Objeto!

✕ models.py



Save

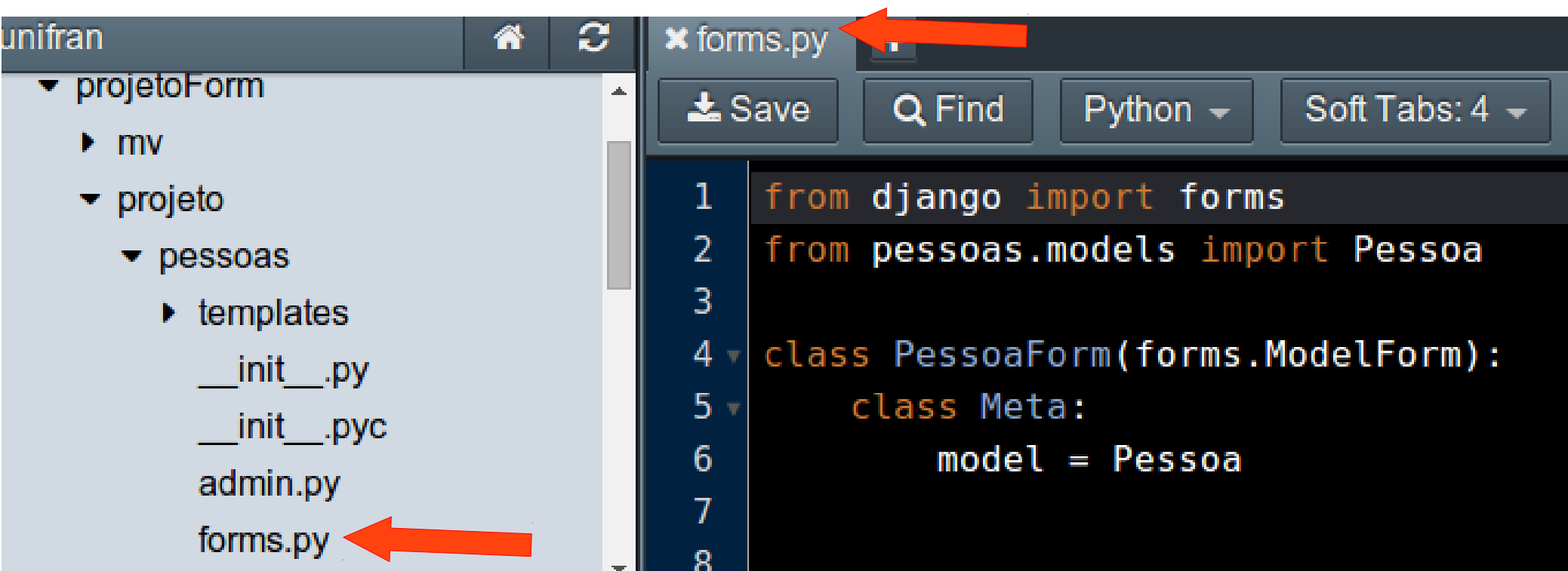
Find

Python ▼

Soft Tabs: 4 ▼

```
1 from django.db import models
2
3 class Pessoa(models.Model):
4     nome = models.CharField(max_length=100)
5     email = models.EmailField()
6     senha = models.CharField(max_length=15)
7     ativo = models.BooleanField(default=True)
8
```

# Criando o Form do Objeto!



The screenshot shows a code editor interface. On the left, a file explorer shows the project structure: `projetoForm` (expanded) contains `mv` and `projeto` (expanded). `projeto` contains `pessoas` (expanded), which contains `templates` and `forms.py`. A red arrow points to `forms.py` in the file explorer. The main editor window shows the code for `forms.py`. A red arrow points to the `forms.py` tab in the editor's tab bar. The code in the editor is:

```
1 from django import forms
2 from pessoas.models import Pessoa
3
4 class PessoaForm(forms.ModelForm):
5     class Meta:
6         model = Pessoa
7
8
```

# Criando o URL e Views!

✕ urls.py

Save

Find

Python

Soft Tabs: 4

```
1 from django.conf.urls import patterns, include, url
2
3 urlpatterns = patterns('',
4     url(r'^$', 'pessoas.views.index'),
5     url(r'^validar/$', 'pessoas.views.validar'),
6 )
```

✕ views.py

Save

Find

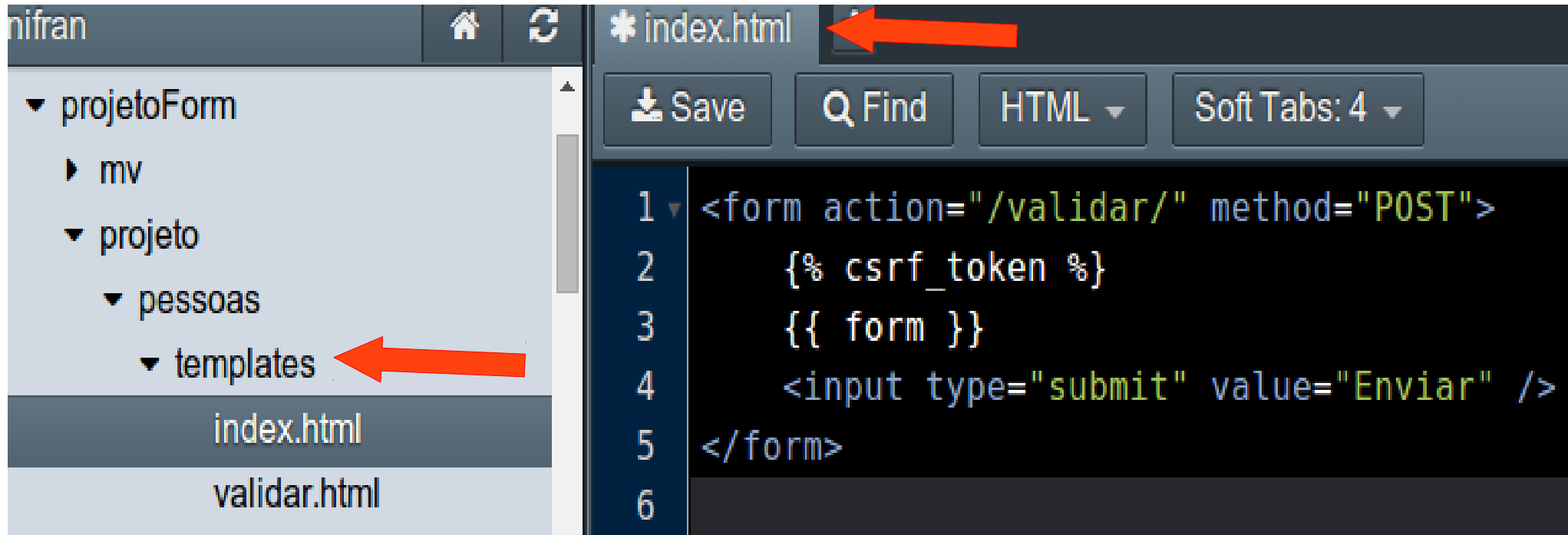
Python

Soft Tabs: 4

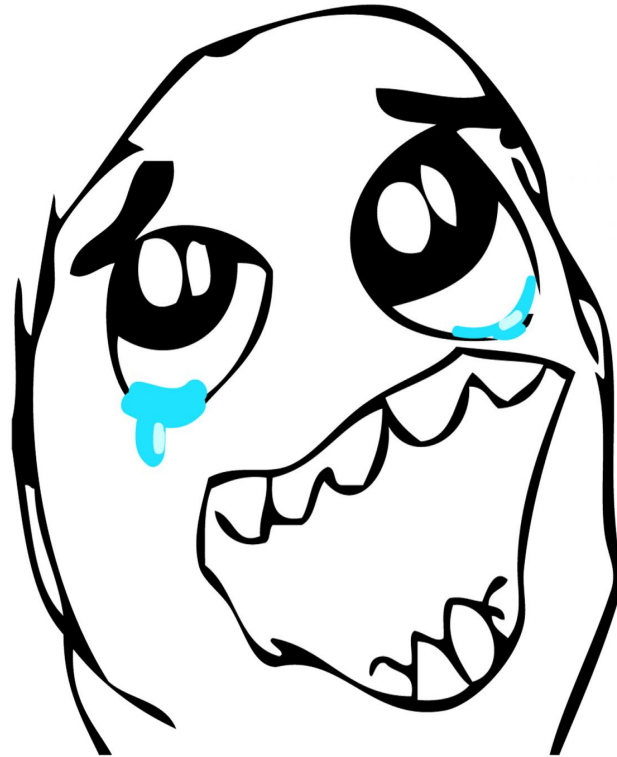
```
1 from django.shortcuts import render
2 from pessoas.forms import PessoaForm
3
4 def index(request):
5     form = PessoaForm()
6     return render(request, 'index.html', {'form': form})
7
```



# Template simples!



# Rodando!



[ultrad.com.br](http://ultrad.com.br)

```
python manage.py runserver 0.0.0.0:8080
```

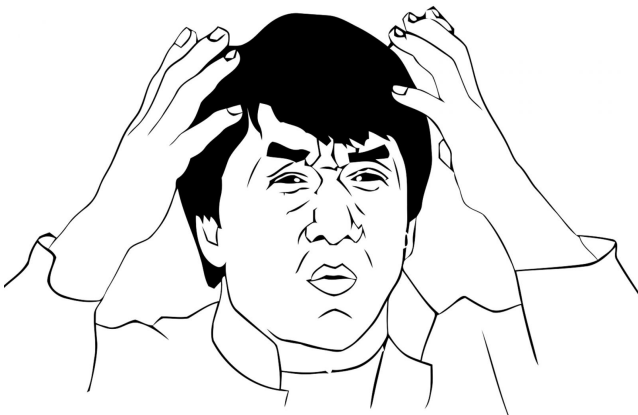
# Elaborando o recebimento!

## Epa, espera aí! A senha não está legal!!!

```
senha = forms.CharField(widget=forms.PasswordInput)
```

ultrad.com.br

Agora sim!!!!



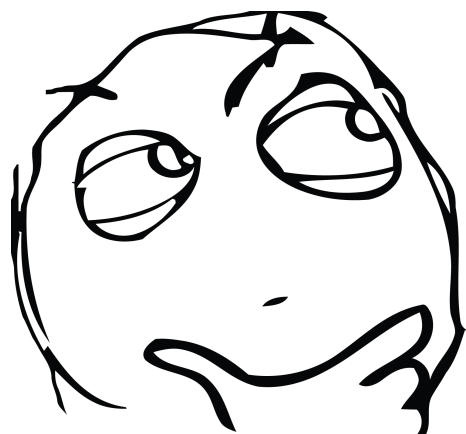


```
def validar(request):  
    if request.method == 'POST':  
        form = PessoaForm(request.POST)  
  
        if form.is_valid():  
            pessoa = Pessoa(**form.cleaned_data)  
            pessoa.save()  
  
            pessoas = Pessoa.objects.all()  
  
            return render(request, 'validar.html', {'form': form, 'pessoas': pessoas})
```

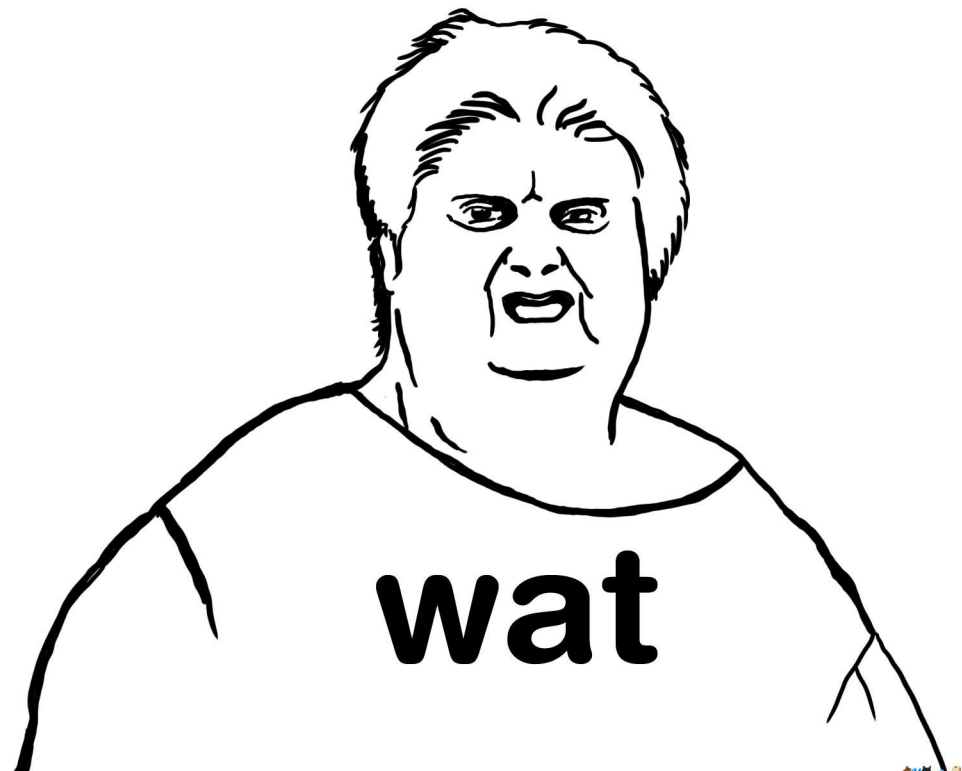
\* validar.html +

Save Find HTML Soft Tabs: 4

```
1 Ok! Validado!<br/>
2
3 FORM:<br/>
4 {{ form.nome.label }} {{ form.nome.value }}<br/>
5 {{ form.email.label }} {{ form.email.value }}<br/>
6 <br/>
7 {{ form.nome }}<br/><br/>
8
9
10 {% for pessoa in pessoas %}
11 - {{pessoa.nome}} {{pessoa.email}}<br/>
12 {% endfor %}
```



Mas e se eu desejo fazer um form sem  
objeto?



\* forms.py

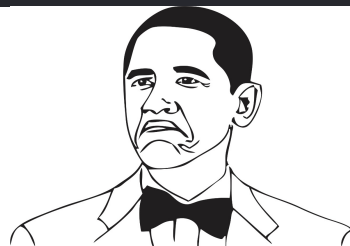
 Save

Q Find

Python ▾

Soft Tabs: 4 ▾

```
1 from django import forms
2 from pessoas.models import Pessoa
3
4 class PessoaForm(forms.ModelForm):
5     class Meta:
6         model = Pessoa
7
8 class PessoaLogin(forms.Form):
9     username = forms.CharField(max_length='30', required=True)
10    password = forms.CharField(widget=forms.PasswordInput, required=True)
11
```

**NOT BAD**

## Dicas:

Pegar dados do form: `Form.data['campo']`

Nomenclatura dos campos: `Models`

Alinhas vertical: `{{ form.as_p }}`

Pegar erros:

```
{% if form.errors %}  
    {{ form.errors.as_text }}  
{% endif %}
```

OU Erros em branco

```
{{ form.non_field_errors }}
```

OU erros individuais

```
{{ form.CAMPO.errors }}
```

