# Django

forms, model-forms

# План

- формы и валидация
- сохранение данных
- model forms
- edit item
- add item
- Шаблоны: понятие контекста и контекст-процессор

Дока наше все!

https://docs.djangoproject.com/en/1.7/topics/forms/

# django.forms

```
from django import forms

class StudentForm(forms.Form):
    PACKAGE_CHOICES = (
        ('standart', 'Standart'),
        ('gold', 'Gold'),
        ('vip', 'VIP'),
        )
    student_name = forms.CharField(max_length=100)
    student_package = forms.ChoiceField(
        choices=PACKAGE_CHOICES,
        widget=forms.RadioSelect)
    student_note = forms.CharField(widget=forms.Textarea)
```

# django form validation

```
def student_edit(request):
    if request.method == ‘post’:
        form = StudentForm(request.POST)
        if form.is_valid():
            #process data
            print form.cleaned_data
            return redirect(‘student_list’)
    else:
        form = StudentForm()

    return render(request, 'students/edit.html',
                                {'form': form})
```

# django form initial

```
def student_edit(request, student_id):
    student = Student.objects.get(id=student_id)
    if request.method == 'post':
        form = StudentForm(request.POST)
        if form.is_valid():
            #process data
            print form.cleaned_data
            return redirect('student_list')
    else:
        form = StudentForm(initial={'student_name': student.name,
                                    'student_package': student.package})

    return render(request, 'students/edit.html',
                           {'form': form})
```

# django form save data

```
def student_edit(request, student_id):
    student = Student.objects.get(id=student_id)
    if request.method == 'post':
        form = StudentForm(request.POST)
        if form.is_valid():
            student.name = form.cleaned_data['student_name']
            student.package = form.cleaned_data['student_package']
            student.save()
            return redirect('student_list', student.id)
    else:
        form = StudentForm(initial={'student_name': student.name,
                                    'student_package': student.package})

    return render(request, 'students/edit.html',
                                {'form': form})
```

# django form template

edit.html
<html>

<form method='post'>
{% csrf_token %}
{{ form }}
<input type='submit' />
</form>

</html>

# django form template

```
<form method='post'>
{% csrf_token %}

{% for field in form %}
    {{ field }}
{% endfor %}

<input type='submit' />

</form>
```

Дока наше все!

https://docs.djangoproject.com/en/1.7/topics/forms/modelforms/

# djangoforms for model

```
from django import forms

class StudentForm(forms.ModelForm):
    class Meta:
        model = Student
```

# django.forms for model

```
from django import forms

class StudentForm(forms.ModelForm):
    class Meta:
        model = Student
        fields = ['name', 'package']
```

# django model form view

```
def student_edit(request, student_id):
    student = Student.objects.get(id=student_id)
    if request.method == 'post':
        form = StudentForm(request.POST, instance=student)
        if form.is_valid():
            student = form.save()
            return redirect('student_list', student.id)
    else:
        form = StudentForm(instance=student)

    return render(request, 'students/edit.html',
                           {'form': form})
```

# django model form customisation

```
from django import forms

class StudentForm(forms.ModelForm):
    class Meta:
        model = Student
        # fields = ['name', 'package']
        exclude = ['note']
        widgets = {
            'package': forms.RadioSelect
            }
```

# django model form customisation

```
from django import forms

class StudentForm(forms.ModelForm):
    class Meta:
        model = Student
        fields = [‘name’, ‘package’]
        labels = {
            'name':  ‘Student name'}
        help_texts = {
            'name': 'Enter first name' }
```

# django model form view add item

```
def student_add(request):
    title = "Student add item"
    if request.method == 'POST':
        form = StudentModelForm(request.POST)
        if form.is_valid():
            student = form.save()
            return redirect('student-edit', student.id)
    else:
        form = StudentModelForm()
    return render(request, ‘students/edit.html',
                  {'form': form, 'title': title})
```

Дока наше все!

https://docs.djangoproject.com/en/1.7/ref/settings/#template-context-processors

# context-processors

django.core.context_processors.media

```
def media(request):
    return {'MEDIA_URL': settings.MEDIA_URL}
```

# RequestContext

```
from django.http import HttpResponse
from django.template import RequestContext,
loader

def my_view(request):
    t = loader.get_template('index.html')
    c = RequestContext(request, {'foo': 'bar'})
    return HttpResponse(t.render(c))
```

# render_to_response

```
from django.template import RequestContext
from django.shortcuts import render_to_response

def my_view(request):
    return render_to_response('index.html',
        {'foo': 'bar'},
        context_instance=RequestContext(request))

def my_view(request):
    return render_to_response('index.html',
        context_instance=RequestContext(request, {'foo': 'bar'}))
```

# render

```
from django.shortcuts import render

def my_view(request):
    return render(request, ’my_template.html',
{'foo': ‘bar'})
```