mercurytide

DJANGO CHEAT SHEET

Version 1.

Talented developers apply at www.mercurytide.co.uk/careers

Template tags

Built in	comments library	
{% autoescape on off %} — {% endautoescape %} {% block name %} — {% endblock %} {% comment %} — {% endcomment %} {% csrf_token %} {% cycle "even" "odd" [as name [silent]] %} {% cycle name %} {% debug %}	{% comment_form_target %} {% get_comment_count for var as name %} {% get_comment_form for var as name %} {% get_comment_list for var as name %} {% get_comment_permalink comment [format_string] %} {% render_comment_form for var %} {% render_comment_list for var %}	
{% extends template %}	flatpages library	
{% filter filter1 filter2 %} {% firstof var1 var2 default %}	{% get_flatpages [url_prefix] as name %}	
Does not auto-escape	i18n <i>library</i>	
{% for var[] in list %} — [{% empty %}] — {% endfor %} forloop.counter forloop.revcounter forloop.revcounter0 forloop.first forloop.parentloop forloop.last {% if expression %} — [{% elif expression %}] — [{% else %}] — {% endif %} Comparisons: == != << => >= in not in Boolean: and or not {% ifchanged [var1] [var2] %} — {% endifchanged %} {% include template [with name = value [only]] %} {% load library %} {% load tag from library %} {% regroup list by attr as name %}	{% blocktrans [context context_name] [with name = value] %} — {% endblocktrans %} {% blocktrans count name = value [context, with etc.] %} — {% plural %} — {% endblocktrans%} {% get_available_languages as LANGUAGES %} {% get_current_language as LANGUAGE_CODE %} {% get_current_language_bidi as LANGUAGE_BIDI %} {% get_language_info for "en" as lang %} lang.code lang.name lang.name_local lang.bidi {% get_language_info for list_of_codes as lang %} {% trans string [context context_name] [as name] [noop] %}	
	I10n library	
	{% localize on off %} — {% endlocalize %}	
{% spaceless %} — {% endspaceless %}	static library	
{% ssi path [parsed] %} {% templatetag openblock %} open closeblock variable brace comment {% url "[app:]viewname" [arg] [kwarg=value] [as name] %} {% verbatim [name] %} — {% endverbatim [name] %} {% widthratio value maxvalue maxwidth %} = value / maxvalue * maxwidth {% with name=value[]%} — {% endwith %}	{% get_media_prefix [as name] %} {% get_static_prefix [as name] %} {% static path [as name] %}	
	staticfiles library	
	{% static path [as name] %}	
	tz library	
	{% get_current_timezone [as TIME_ZONE] %} {% localtime on off %} — {% endlocaltime %} {% timezone time zone name %} — {% endtimezone %}	
cache library	webdesign <i>library</i>	
$ \{\% \ \textbf{cache} \ seconds \ name \ [\textit{vary_by} \dots] \ \% \} = \{\% \ \textbf{endcache} \ \% \} $	{% lorem [number] [w p b] [random] %}	
	(% loroni [namber] [w p b] [landoni] %}	

Filters humanize library Text formatting Dates & times Escaping | date: format apnumber lower safe intcomma I time: format upper safeseg title | timesince[: datetime] intword escape capfirst | timeuntil[: datetime] | naturalday[: format] force escape naturaltime slugify escapeis Lists ljust: width ordinal addslashes length rjust: width fix ampersands |length is:length i18n library center: width liriencode first language_name wordwrap: width |urlencode[:safe chars] last language name local wordcount random bidi striptags General | join: string removetags: "p br" I10n library | slice: "1:9" | default: value Case sensitive. |make list default if none: value localize cut: substring unordered list | dictsort: key unlocalize linebreaks | dictsortreversed: key Numbers linebreaksbr tz library pprint linenumbers add: number localtime | stringformat: format | divisibleby: number phone2numeric |timezone: tz name | yesno[: "yes,no[,none]"] truncatechars: count filesizeformat utc truncatewords: count | floatformat[: decimal places] truncatewords html: count get digit: position urlize position 1 = 1's, 2 = 10's

Template date formats

| pluralize[: "y,ies"]

urlizetrunc: length

0 (Sun) to 6 (Sat)

Year		AM & PM		Localized			
y Y	13 2013	a A	a.m. or p.m. AM or PM	SHORT_DATETIME_FORMAT Short date		Long date & time Short date & time	
Mon	th	Ηοι	ır	DATE_FORMAT Long date		· ·	
F E	January (localised full date form)	H G	00 to 23 0 to 23			Short date Time	
	M Jan b jan N Jan. (A.P.) n 1 to 12 m 01 to 12	h	01 to 12	Tin	nezone		
		g	1 to 12	T	TZ code: EST, CET		
		Minutes		e TZ name: GMT, -500, US/Eastern			
		i	00 to 59	0	0 TZ offset: +0200		
		Seconds		Z TZ offset (sec): -43200 to 43200			
		s	00 to 59	Mis	sc		
Day	Day μs			I DST in effect: 1 or 0			
d j	01 to 31 1 to 31	u	000000 to 999999	L	Is leap year: True or False		
		Time		z Day of year: 0 to 365			
S	suffix: st, nd, rd or th	f P	1, 1:30 1 a.m. 1 a.m. 1:30 p.m., midnight , noon	t Days in month: 28 to 31 W Week number: 1 to 53 Week-numbering year: 2013			
ı	Friday Standards						
D	Fri	c ISO 8601: 2013-03-28T09:47:23.538501+02:00					

RFC 2822: Thu, 21 Dec 2000 16:01:07 +0200

U Unix time: 1364471243

Version 1.5

[allow empty file=False]

[allow empty file=False]

[path=directory path]

[recursive=False]

[allow files=True]

[allow folders=False]

[min length=number]

[max length=number]

[min length=number]

[max length=number]

[unpack ipv4=False]

[fields=tuple]

[fields=tuple]

[protocol="both"|"IPv4"|"IPv6"]

invalid list

[match=regex]

[max_length=number] (of filename)

Form fields

Dates & Times Common kwargs DateField [input formats=iterable] required=True TimeField [input formats=iterable] label=string **DateTimeField** [input formats=iterable] help text=string **SplitDateTimeField** [input date formats=iterable] initial=value or function [input time formats=iterable] validators=list of validators widget=Widget class Numbers error messages=dict IntegerField [max value=int] localize=False [min value=int] (e.g. localize dates, decimals) FloatField [max value=float] [min value=float] Boolean DecimalField [max digits=number] **BooleanField** [decimal places=number] NullBooleanField [max value=decimal] [min value=decima/] Strings Files CharField [min length=number] FileField [max length=number] (of filename)

ImageField

FilePathField

Internet

EmailField

URLField

(IPv4/IPv6)

ComboField

MultiValueField

Misc

IPAddressField (IPv4)

GenericIPAddressField

[max length=number]

SlugField [min length=number] [max length=number]

RegexField regex=regex

[min length=number] [max length=number]

[cache choices=False]

[cache choices=Fa/se]

queryset=queryset

Choices

ChoiceField choices=iterable **TypedChoiceField** choices=iterable [coerce=function] [empty value=""] MultipleChoiceField choices=iterable **TypedMultipleChoiceField** choices=iterable [coerce=function] [empty value=""] ModelChoiceField quervset=quervset [empty label="----"]

ModelMultipleChoiceField error messages keys

required min_length invalid_date max length invalid time invalid choice min value

invalid (int, email, etc.)

max value max digits max decimal places max whole digits

missing (file) empty (file) invalid image invalid pk value class Media:

css = {"all | screen | print": (filenames), ... } **is** = (filenames) [extend = True]

Form & widget Media

Validators

MaxLengthValidator(length) MinLengthValidator(length) validate email validate slug validate url RegexValidator([regex=""]. [message="Enter a valid value"]. [code="invalid"]) MaxValueValidator(value) MinValueValidator(value) validate ipv4 address validate ipv6 address validate ipv46 address

validate comma separated integer list

Formsets

FormSetClass = formset factory(form, [extra=1.] [max num=None]. [can order=False], [can delete=False]. [formset=BaseFormSet])

FormSetClass(data_[files_] [initial=list_] [prefix=string])

Formset attributes

formset.empty form formset.management form formset.ordered forms

Form attributes

form.ORDER form **DELETE**

Widgets

General	
TextInput PasswordInput HiddenInput TextArea	[format=date_format]
Dates & times	
DateInput	[format=date_format]

DateTimeInput [format=datetime format] TimeInput [format=time format] SplitDateTimeWidget [date format=date format [time format=time format]

SplitHiddenDateTimeWidget

SelectDateWidget [**years**=iterable]

Choices

Select [choices=iterable] **NullBooleanSelect** [choices=((1, "Unknown"). (2, "Yes"), (3, "No"))] RadioSelect [choices=iterable] SelectMultiple [choices=iterable] CheckboxSelectMultiple [choices=iterable] MultipleHiddenInput [choices=iterable]

Files

FileInput ClearableFileInput

Common kwargs

attrs=dict eg.{"class": "foo", "rows": "5"}

ModelForms & ModelFormSets

ModelForm Meta attributes		Factories	
abstract=False app_label=string db_table=string model=model fields=tuple	exclude=tuple widgets=dict { "field": TextInput() }	modelform_factory(model, [form=FormClass,] [fields=tuple.] [exclude=tuple.] [widgets=dict]) modelformset_factory(model, queryset=queryset.] [formset=BaseModelFormSet,]) inlineformset_factory(parent_model, child_model, [fk_name=field.] [can_delete=True.]	
		[formset=BaseInlineFormSet,])	

DJANGO CHEAT SHEET

		Internet		
Model fields sh	owing defaults	EmailField URLField	[max_length=75] [max_length=200]	
null=False blank=False choices=iterable	help_text=string primary_key=False unique=False	IPAddressField (IPv4) GenericIPAddressField	[protocol="both" "IPv4" "IPv6"] [unpack_ipv4=False]	
db_column=name	unique_for_date=date_field_name	Files		
db_index=False default=value_or_function editable=True error_messages=dict	unique_for_month=date_field_name unique_for_year=date_field_name verbose_name=string validators=iterable	FileField ImageField	upload_to=path_or_function [storage=storage] [max_length=number] upload_to=path_or_function	
Numbers		illugor ioia	[storage=storage]	
AutoField Booleans BooleanField	primary_key=True (mandatory)	FilePathField	[height_field=field_name] [width_field=field_name] [max_length=number] path=path	
NullBooleanField Numbers IntegerField (>= 32 bit) PositiveIntegerField (>= 32 bit) SmallIntegerField (DB specific size			<pre>[match=regex] [recursive=False] [allow_files=True] [allow_folders=False] [max_length=100]</pre>	
PositiveSmallIntegerField (DB s	pecific size)	Relationships		
BigIntegerField (64 bit) DecimalField FloatField	max_digits=number decimal_places=number	ForeignKey	model (or "self") [limit_choices_to=dict_or_Q] [related_name=string] [to_field=field_name]	
Strings		OneToOneField	[on_delete=action] model (or "self")	
CharField TextField SlugField CommaSeparatedIntegerField	max_length=number [max_length=50] max_length=number	ManyToManyField	[limit_choices_to=dict_or_Q] [related_name=string] [to_field=field_name] [on_delete=action]	
Strings			[parent_link=False]	
CharField TextField	max_length=number	error_messages keys		
SlugField CommaSeparatedIntegerField	[max_length=50] max_length=number	null blank unique		
Dates & times		invalid invalid choice		
DateField DateTimeField	[auto_now=False] [auto_now_add=False] [auto_now=False]	invalid_date on_delete actions		
TimeField	[auto_now_add=False] [auto_now=False] [auto_now_add=False]	CASCADE Protect Set_null	SET_DEFAULT SET(value_or_function) DO_NOTHING	

Model Meta attributes abstract=False ordering=tuple app_label=string permissions=tuple db_table=string **proxy**=False db tablespace=string unique_together=tuple get latest by=field name index_together=tuple managed=True verbose_name=string order_with_respect_to=field_name verbose_name_plural=string

Validators

MaxLengthValidator(length)	MaxValueValidator(value)
MinLengthValidator(length)	MinValueValidator(value)
validate_email	validate_ipv4_address
validate_slug	validate_ipv6_address
validate_url	validate_ipv46_address
$\textbf{RegexValidator}([\textbf{regex}=^{""}],[\textbf{message}=$	validate_comma_separated_integer_list
"Enter a valid value"], [code="invalid"])	

QuerySets

Methods that return QuerySets	QuerySet & Queries	
all() none() filter(query) exclude(query) order_by(field,) reverse() annotate(aggregate_query) distinct([field,]) values(field,) values_list(field, [flat=False]) dates(field, kind="year month day",	exact=value None iexact=value contains=string icontains=string startswith=string istartswith=string endswith=string iendswith=string regex=pattern iregex=pattern gt=value gte=value	It=value Ite=value in=iterable queryset range=(start, end) year=year month=1-12 day=1-31 week_day= 1 (Sun) - 7 (Sat) isnull=boolean search= mysql fulltext query
	Aggregation function Count(field, [distinct=Fal Sum(field) Max(field) Min(field) Avg(field) StdDev(field, [sample=Fal Variance(field, [sample=	se]) alse])

Other methods

$\begin{array}{l} \textbf{aggregate}(aggregate_query) \rightarrow \textit{dict} \\ \textbf{update}(\textit{field} = \textit{value},) \end{array}$
delete()
in_bulk(ids) → instances dict, keyed by id
<pre>bulk_create(instances, [batch_size=number])</pre>
$iterator() \rightarrow iterator$

DJANGO CHEAT SHEET

ModelAdmin

List columns

list_display = field_names list_display_links = field_names list_editable = field_names list_filter = field_names search_fields = field_names date_hierarchy = field_name

List actions

actions = list
actions_on_top = True
actions_on_bottom = False
actions_selection_counter = True
ordering = field_names

List options

ordering = field_names list_select_related = False list_per_page = 100 list_max_show_all = 200 paginator = paginator_class

Form contents

Form contents

Form layout

save on top = Fa/se

add_form_template = template
change_form_template = template
change_list_template = template
delete_confirmation_template = template
delete_selected_confirmation_template = template
object_history_template = template

Templates

add_form_template = template
change_form_template = template
change_list_template = template
delete_confirmation_template = template
delete_selected_confirmation_template = template
object_history_template = template

Queryset

queryset(self, request) → *queryset*

Saving

```
save_model(self, request, obj, form, is_change) - must call obj.save()
delete_model(self, request, obj) - must call obj.delete()
save_formset(self, request, form, formset, is_change) - must call formset.save()
save_related(self, request, form, formsets, is_change) - must call save() for each formset
```

Permissions

has_add_permission(self, request) → boolean
has_change_permission(self, request, obj=None) → boolean
has_delete_permission(self, request, obj=None) → boolean

URLs

get_urls(self) → urlpatterns

Change List

```
get_ordering(self, request) → field_names
get_list_display(self, request) → field_names
get_list_display_links(self, request, list_display) → field_names
get_list_filter(self, request) → field_names
get_changelist(self, request, **kwargs) → ChangeList_class
get_changelist_form(self, request, **kwargs) → ModelForm_class
get_changelist_formset(self, request, **kwargs) → ModelFormSet_class
get_paginator(self, request, queryset, per_page, orphans=0,
allow_empty_first_page=True) → paginator
```

Add/Change Form

 $\begin{tabular}{ll} \textbf{get_readonly_fields} (self, request, obj=None) \rightarrow field_names \\ \textbf{get_prepopulated_fields} (self, request, obj=None) \rightarrow prepopulated_fields_dict \\ \textbf{get_fieldsets} (self, request, obj=None) \rightarrow fieldsets_tuple \\ \textbf{get_form} (self, request, obj=None, **kwargs) \rightarrow form_class \\ \textbf{get_inline_instances} (self, request, obj=None) \rightarrow inline_instances_tuple \\ \textbf{get_formsets} (self, request, obj=None) \rightarrow pields InlineModelAdmins \\ \textbf{formfield_for_choice_field} (self, db_field, request, **kwargs) \rightarrow field \\ \textbf{formfield_for_foreignkey} (self, db_field, request, **kwargs) \rightarrow field \\ \textbf{formfield_for_manytomany} (self, db_field, request, **kwargs) \rightarrow field \\ \textbf{formfield_for_manytomany} (self, db_field, request, **kwargs) \rightarrow field \\ \end{tabular}$

InlineModelAdmin

Standard subclasses

StackedInline TabularInline

Attributes & methods

model = model (mandatory)
fk_name = field
form = ModelForm
formset = BaseInlineFormSet
extra = 1
max_num = number

verbose_name = string
verbose_name_plural = string
can_delete = True
template = template
get_formset(self, request, obj = None) →
BaseInlineFormSet_class

AdminSite

```
login_form = form
index_template = template
login_template = template
logout_template = template
password_change_template = template
password_change_done_template = template
```

mercurytide

Mercurytide is built upon a foundation of great people. We love all things digital, and our clients benefit from our full-service delivery - from strategy and thorough production to ongoing support. In order to keep up with the demand for our expert services, Mercurytide is growing.

You will join a culture of exceptional people, who pride themselves on being up to speed with the latest technologies. Our forward-thinking staff have created bespoke solutions for a wide variety of clients, from local retailers to global market leaders. We champion the strengths of our staff and fuel progress by learning from each other.

Life at Mercurytide is never dull; we have a busy calendar of work, plus social events for our staff. If you are talented and think that your skill set can benefit our clients, then drop us a line today.

Don't forget be our friend on Facebook and link up with us on LinkedIn

Talented developers apply at www.mercurytide.co.uk/careers