Django update view

Django Update View[¶](https://learnbatta.com/course/django/update-view/#django-update-view)

* Django update view is used to update a single record of the database table.
* It shows an editable form to the user with the current row data.
* User can make the changes to the form and on submit it then it will save the data to the database table.

Contact Model[¶](https://learnbatta.com/course/django/update-view/#contact-model)

my\_app/models.py

from django.db import models

class Contact(models.Model):

first\_name = models.CharField(max\_length=30)

last\_name = models.CharField(max\_length=30)

email = models.EmailField(max\_length=255)

phone = models.CharField(max\_length=10, null=True)

class Meta:

db\_table = "contact"

form template[¶](https://learnbatta.com/course/django/update-view/#form-template)

templates/form\_template.py

<form action="" method="POST" enctype="multipart/form-data">

{% csrf\_token %}

{{ form.as\_table }}

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

</form>

function based update view[¶](https://learnbatta.com/course/django/update-view/#function-based-update-view)

* It's simple function handles both GET and POST requests
  + GET - displays the form to the user.
  + POST - recieves the user submitted info
* Let's looke at code below.

from django.http import HttpResponse

from django.shortcuts import render, get\_object\_or\_404

from my\_app.models import Contact

from my\_app.forms import ContactModelForm

def fbv\_update\_view(request, pk):

obj = get\_object\_or\_404(Contact, id=pk)

if request.method.lower() == 'post':

form = ContactModelForm(request.POST, instance=obj)

if form.is\_valid():

form.save()

return HttpResponse("<h1>Updated</h1>")

else:

form = ContactModelForm(instance=obj)

context = {'form': form}

return render(request, 'form\_template.html', context)

* get\_object\_or\_404 gets the object from the db if found otherwise throws HTTP 404 error.
* For GET request
  + render the html template with form
  + ContactModelForm(instance=obj) loads the initial data
* For POST request
  + validates the form ContactModelForm(request.POST, instance=obj)
  + save the form if it's valid otherwise renders the form template with errors.

class based update view[¶](https://learnbatta.com/course/django/update-view/#class-based-update-view)

* Django provides a generic view UpdateView to update the table row.

from django.http import HttpResponse

from django.views.generic.edit import UpdateView

from my\_app.models import Contact

from my\_app.forms import ContactModelForm

class CBVUpdateView(UpdateView):

form\_class = ContactModelForm

template\_name = 'form\_template.html'

pk\_url\_kwarg = 'pk'

model = Contact

def form\_valid(self, form):

form.save()

return HttpResponse("<h1>Updated</h1>")

* Django provides the generic view class UpdateView.
* It contains the inbuilt funcitonality which we implemented in the function based view.
* In order to work with UpdateView we need to provider attributes form\_class, template\_name, pk\_url\_kwarg and model or methods get\_object() or get\_queryset()
  + template\_name - used to render the template
  + pk\_url\_kwarg - used to identify the object in the database
  + model or get\_object() or get\_queryset() - used to get the object using pk\_url\_kwarg

configure the urls[¶](https://learnbatta.com/course/django/update-view/" \l "configure-the-urls" \o "Permanent link)

* open my\_app/urls.py and add below code to it.

from django.urls import path

from my\_app import views

urlpatterns = [

# ...

path('fbv\_update\_view/<int:pk>', views.fbv\_update\_view, name='fbv\_update\_view'),

path('cbv\_update\_view/<int:pk>', views.CBVUpdateView.as\_view(), name='cbv\_update\_view'),

# ...

]

* Open url <http://127.0.0.1:8000/my_app/fbv_update_view/1> to update the object using function based update view.
* Open url <http://127.0.0.1:8000/my_app/cbv_update_view/1> to update the object using class based update view.
* Replace the 1 with any id in the contact table.
* It throws 404 status code if id does not exist in the table.

References[¶](https://learnbatta.com/course/django/update-view/#references)

* <https://docs.djangoproject.com/en/dev/ref/class-based-views/generic-editing/#django.views.generic.edit.UpdateView>
* <https://ccbv.co.uk/projects/Django/4.1/django.views.generic.edit/UpdateView/>