

First of all install `django` for this open command prompt and go to d: drive and type following commands :-

Step 1:-

```
D:\>python -m venv env
```

Step 2:-

```
D:\>env\Scripts\activate
```

Step 3:-

```
D:\>pip install django
```

To create a Django application that performs "insert from html form to database"

1. Create a Project

Use command `django-admin startproject crudexample`

As shown below in command prompt

```
C:\Users\software>d:  
D:\>django-admin startproject crudexample
```

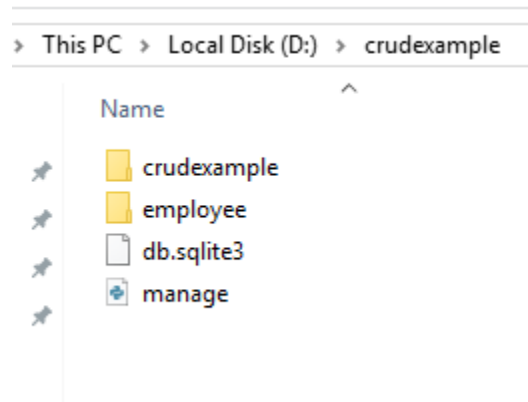
2. Create an App

Use command `python manage.py startapp employee`

As shown below.

```
D:\>cd crudexample
D:\crudexample>python manage.py startapp employee
```

Now open your project folder you will see structure look like this



Now under `employee` folder create `templates` folder as shown below

PC > Local Disk (D:) > crudexample > employee

Name	Date
__pycache__	10/2
migrations	10/2
templates	10/2
__init__	10/2
admin	10/2
apps	10/2
models	10/2
tests	10/2
views	10/2

and under **templates** folder create index.html file as shown below

› This PC > Local Disk (D:) > crudexample > employee > templates

Name	Date modified
index	10/27/2022 2:35 PM

Code for index.html file:-

```
<html>
  <head>

  </head>
  <body>
    <h1> hello world</h1>
  </body>
</html>
```

Now open settings.py file of project folder (**crudexample**)

:-

Add **'employee'** (app name) as shown below

```
# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'employee'
]
```

Now open `views.py` file of your **employee** (app folder) type following code :-

```
from django.shortcuts import render

# Create your views here.

def emp(request):
    return render(request, 'index.html')
```

now open `urls.py` file of your project folder (**crudexample**) type following code:-

```
from django.contrib import admin
from django.urls import path
from employee import views

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', views.emp)
]
```

Now runs your project as shown below:-

```
C:\Users\software>d:
D:\>cd crudexample
D:\crudexample>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).

You have 18 unapplied migration(s). Your project may not work properly without these migrations:
  auth, contenttypes, sessions.
Run 'python manage.py migrate' to apply them.
October 27, 2022 - 14:50:58
Django version 4.1, using settings 'crudexample.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Now start step by to code insert, update, delete in django with database:-

Step 1:-

Write code for `models.py` file (under your app folder (employee):-

```
from django.db import models

# Create your models here.
class Members(models.Model):
    firstname = models.CharField(max_length=255)
    lastname = models.CharField(max_length=255)
```

Step 2:-

Write code for `admin.py` file to register model:-

```
from django.contrib import admin

from employee.models import Members

# Register your models here.
admin.site.register(Members)
```

Step 3:-

run in following command in your command prompt:-

```
py manage.py makemigrations
```

step 4:-

write code for urls.py file which is under your project folder (crudexample) :-

```
from django.contrib import admin
from django.urls import include, path

urlpatterns = [
    path('employee/', include('employee.urls')),
    path('admin/', admin.site.urls),
]
```

Step 5:-

Write code for urls.py file which is under your app folder (employee):-

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

urlpatterns = [
    path('', views.index, name='index'),

    path('add/', views.add, name='add'),
    path('add/addrecord/', views.addrecord, name='addrecord'),
    path('delete/<int:id>', views.delete, name='delete'),
    path('update/<int:id>', views.update, name='update'),
    path('update/updaterecord/<int:id>', views.updaterecord, name='updaterecord'),
```

```
]
```

Step 6:-

Write code for `views.file` under your app folder `(employee)`:-

```
from django.shortcuts import render
from .models import Members
from django.http import HttpResponse, HttpResponseRedirect
from django.template import loader
from django.urls import reverse

# Create your views here.

def emp(request):
    return render(request, 'index.html')
def add(request):
    return render(request, 'add.html')

def addrecord(request):
    x = request.POST['first']
    y = request.POST['last']
    member = Members(firstname=x, lastname=y)
    member.save()
    data="inserted"
    return render(request, 'add.html', {'data':data})

def index(request):
    mymembers = Members.objects.all().values()
    template = loader.get_template('index.html')
    context = {
        'mymembers': mymembers
    }
    return HttpResponse(template.render(context, request))

def update(request, id):
    mymember = Members.objects.get(id=id)

    context = {
        'mymember': mymember,
    }
}
```

```

    return render(request, 'update.html', context)

def updaterecord(request, id):
    first = request.POST['first']
    last = request.POST['last']
    member = Members.objects.get(id=id)
    member.firstname = first
    member.lastname = last
    member.save()
    return HttpResponseRedirect(reverse('index'))

def delete(request, id):
    member = Members.objects.get(id=id)
    member.delete()
    return HttpResponseRedirect(reverse('index'))

```

Step 7:- write code for `index.html` file code under templates folder of your app folder (`employee`):-

```

<h1>Members</h1>

<table border="1">
{% for x in mymembers %}
<tr>
<td><a href="update/{{ x.id }}">{{ x.id }}</a></td>
<td>{{ x.firstname }}</td>
<td>{{ x.lastname }}</td>
<td><a href="delete/{{ x.id }}">delete</a>
</tr>
{% endfor %}
</table>

<p>
<a href="add/">Add member</a>
</p>

```

write code for **add.html** file code under templates folder of your app folder (**employee**):-

```
<h1>Add member</h1>

<form action="addrecord/" method="post">
{% csrf_token %}
First Name:<br>
<input name="first">
<br><br>
Last Name:<br>
<input name="last">
<br><br>
<input type="submit" value="Submit">
</form>
{{data}}
```

Write code for **update.html** file code under templates folder of your app folder (**employee**):-

```
<h1>Update member</h1>

<form action="updaterecord/{{ mymember.id }}" method="post">
{% csrf_token %}
First Name:<br>
<input name="first" value="{{ mymember.firstname }}">
<br><br>
Last Name:<br>
<input name="last" value="{{ mymember.lastname }}">
<br><br>
<input type="submit" value="Submit">
</form>
```

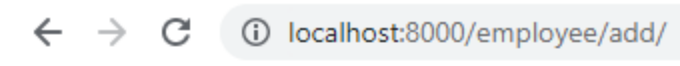
Output:-

Open command prompt and run command

```
py manage.py runserver
```

and open any browser and to add record

type path **localhost:8000/employee/add**



Add member

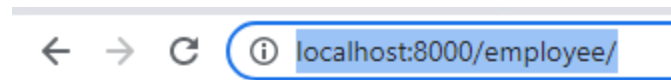
First Name:

Last Name:

Submit

To display record type path:-

<http://localhost:8000/employee/>



Members

2	om	sir testing	delete
3	ram	ram testing	delete
5	nirmala	ji	delete
6	nirmala	sir testing	delete

[Add member](#)