

Django insert, update, delete, display examples with mysql database:-

```
D:\> mkdir myap
```

```
D:\myap>cd myap
```

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

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

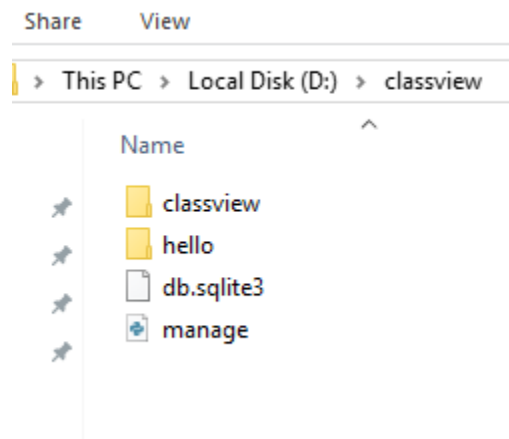
```
D:\myap>pip install django
```

```
D:\myap>django-admin startproject classview
```

```
D:\myap>cd classview
```

```
D:\myap\classview>python manage.py startapp hello
```

After it you will see following structure .

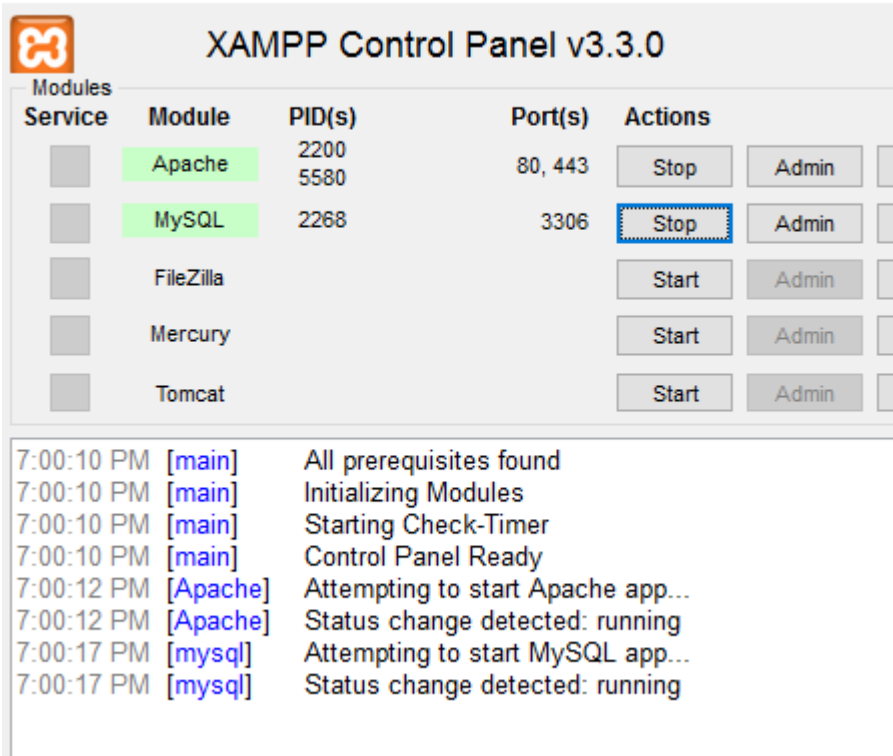


Django MySQL – How to Set up Django to use a Database in xampp?

Step 1:-

run xampp and start apache and mysql as shown below:-

XAMPP Control Panel v3.3.0 [Compiled: Apr 6th 2021]



The screenshot shows the XAMPP Control Panel interface. At the top, it displays the version 'v3.3.0' and the compilation date 'Apr 6th 2021'. Below this is a table of modules with columns for Service, Module, PID(s), Port(s), and Actions. The 'Apache' and 'MySQL' modules are highlighted in green, indicating they are running. The 'MySQL' 'Stop' button is highlighted with a blue dashed border. Below the table is a log window showing the following messages:

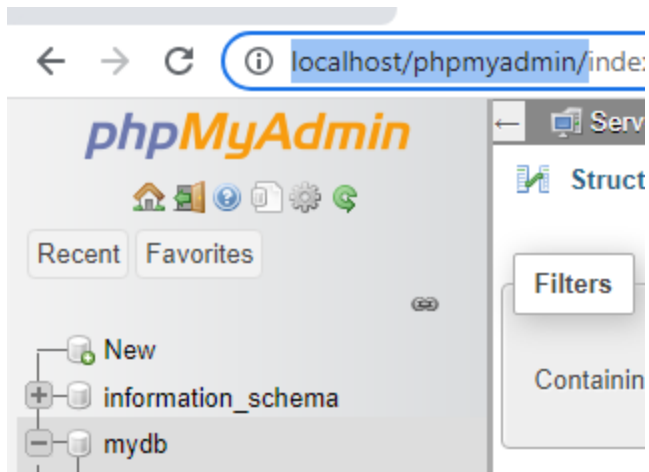
Service	Module	PID(s)	Port(s)	Actions
<input type="checkbox"/>	Apache	2200 5580	80, 443	Stop Admin
<input type="checkbox"/>	MySQL	2268	3306	Stop Admin
<input type="checkbox"/>	FileZilla			Start Admin
<input type="checkbox"/>	Mercury			Start Admin
<input type="checkbox"/>	Tomcat			Start Admin

7:00:10 PM [main] All prerequisites found
7:00:10 PM [main] Initializing Modules
7:00:10 PM [main] Starting Check-Timer
7:00:10 PM [main] Control Panel Ready
7:00:12 PM [Apache] Attempting to start Apache app...
7:00:12 PM [Apache] Status change detected: running
7:00:17 PM [mysql] Attempting to start MySQL app...
7:00:17 PM [mysql] Status change detected: running

Step 2:- open any browser and type <http://localhost/phpmyadmin/>

&

Create database "mydb" in phpmyadmin as shown below :-



Step 3:-

Go to your project folder and in settings.py file make following changes as shown below.

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'mydb',
        'USER': 'root',
        'PASSWORD': '',
        'HOST': 'localhost',
        'PORT': '3306',
    }
}
```

Step 4:- after that run following commands as shown below.

First run command

```
py manage.py makemigrations
```

after it run command

```
py manage.py migrate
```

as shown below:-

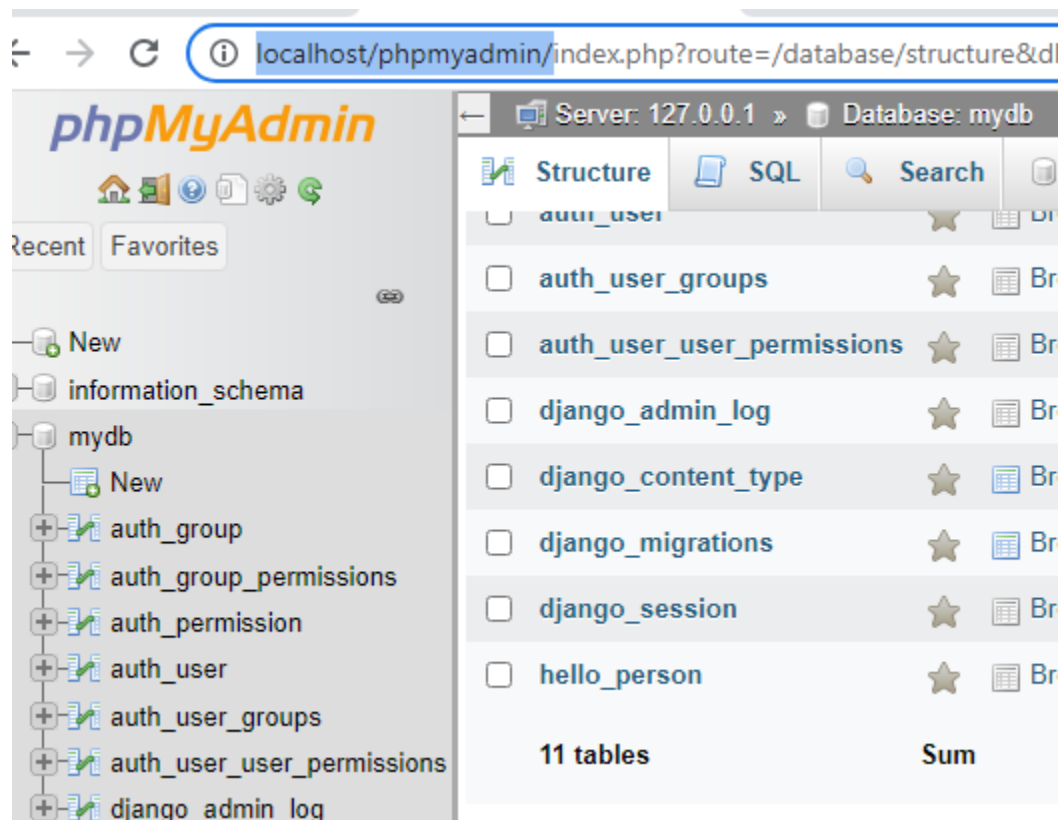
```
D:\classview>py manage.py makemigrations
Migrations for 'hello':
  hello\migrations\0001_initial.py
    - Create model Person

D:\classview>py manage.py migrate
```

After it open your

<http://localhost/phpmyadmin/>

you will see your table is created as shown below.



After it open your **hello(app)** folder and write following code in **models.py** file :-

```

from django.db import models

# Create your models here.

class Person(models.Model):
    first_name = models.CharField(max_length=30)
    last_name = models.CharField(max_length=30)

```

after it open **views.py** file of hello app folder and write following code:-

```

from django.views.generic import TemplateView
from django.http import HttpResponseRedirect, HttpResponseRedirect
from django.template import loader
from django.shortcuts import render

from .models import Person
from django.urls import reverse

def add(request):

    return render(request, 'add.html')

def addrecord(request):
    x = request.POST['first']
    y = request.POST['last']
    member = Person(first_name=x, last_name=y)
    member.save()
    return HttpResponseRedirect("record inserted")

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

def delete(request, id):

```

```

member = Person.objects.get(id=id)
member.delete()
return HttpResponseRedirect(reverse('index'))

def update(request, id):
    mymember = Person.objects.get(id=id)
    template = loader.get_template('update.html')
    context = {
        'mymember': mymember,
    }
    return HttpResponse(template.render(context, request))

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

```

after it open `urls.py` file of your hello app folder and write following code:-

```

from django.urls import path
from hello.views import AboutView, ContactView
from . import views

urlpatterns = [

    path('add/', views.add, name='add'),
    path('add/addrecord/', views.addrecord, name='addrecord'),
    path('', views.index, name='index'),
    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'),

]

```

After it open urls.py file of your project folder and write code:-

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

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

In your hello app folder inside template folder create file add.html :-

```
<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}}
```

And create file update.html :-

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

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

And create index.html file :-

```
<html>
<head>
{% load static %}
<link rel="stylesheet" href="{% static 'myfirst.css' %}">

</head>
<body>
<h1>Members</h1>

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

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

And now run your project by using command `py manage.py runserver` as shown below :-

```
D:\>cd classview
D:\classview>py manage.py runserver
```

<http://127.0.0.1:8000/hello/>