

RecipeProject:

models.py

```
from django.db import models

class RecipeModel(models.Model):
    recipename = models.CharField(max_length=200)
    ingredients = models.CharField(max_length=200)
    making = models.CharField(max_length=200)
    recipepic = models.ImageField(upload_to='images/')
```

admin.py

```
from django.contrib import admin
from .models import RecipeModel

admin.site.register(RecipeModel)
```

forms.py

```
from django import forms
from .models import RecipeModel

class RecipeForm(forms.ModelForm):
    class Meta:
        model = RecipeModel
        fields = "__all__"
```

Views.py:

```
from django.shortcuts import render, redirect, get_object_or_404
from .forms import RecipeForm
from .models import RecipeModel

def home(request):
    data = RecipeModel.objects.all()
    return render(request, 'home.html', {'data': data})

def create(request):
    if request.method == 'POST':
        form = RecipeForm(request.POST, request.FILES)
        if form.is_valid():
            form.save()
            return redirect('home')
    else:
        form = RecipeForm()
        return render(request, "create.html", {'form': form})

def details(request, id):
    data = get_object_or_404(RecipeModel, pk=id)
    return render(request, "details.html", {'data': data})

def delete(request, id):
    data = get_object_or_404(RecipeModel, pk=id)
    data.delete()
    return redirect('/home')
```

Templates:

home.html

```
{% load static %}
<table>
  <thead><tr>
    <th>Name</th>
    <th>Image</th>
    <th>Delete</th>
  </tr></thead>
  <tbody>
    {% for ele in data %}
      <tr>
        <td><a href = "/"{{ ele.id }}">{{ ele.recipeName }}</a></td>
        {% if ele.recipepic %}
          <td><img src = "{{ ele.recipepic.url }}" height="50"
width="50" alt ='img' ></img> </td>
        <td><form method="post" action="{% url 'delete' ele.id %}">
          {% csrf_token %}
          <button type="submit">Delete</button>
        </form></td></tr>
      {% endfor %}
    </tbody>
  </table>
<center>
<a href = "{% url 'create' %}">Create Recipe</a>
</center>
```

create.html:

“”” When you upload file or images enctype = “multipart/form-data”

Must be need “””

```
<form method="post" enctype="multipart/form-data">
  {% csrf_token %}
  {{ form.media }}
  {{ form.as_p }}
  <br>
  <a href="{% url 'home' %}"><input type="submit"
value="submit"></a>
</form>
```

details.html:

```
<h2>Recipe Name : {{ data.recipename }}</h2>
<form method="post">
    <label>Recipe Name</label>{{ data.recipename }}<br />
    <label>Required items</label>{{ data.ingredients }}<br />
    <label>Making Process</label>{{ data.making }}<br />
    <label>recipepic</label>{{ data.recipepic }}<br />
</form>
<br>
    <a href="{% url 'home' %}"><input type="submit"
value="Home"></a>
```

urls.py

```
from django.contrib import admin
from django.urls import path
from recipeapp import views
from django.conf import settings
from django.conf.urls.static import static
app_name = 'recipeapp'

urlpatterns = [
    path('admin/', admin.site.urls),
    path("", views.home, name = 'home'),
    path("home", views.home, name = 'home'),
    path("create", views.create, name = 'create'),
    path('<int:id>/details', views.details, name = 'details'),
    path('<int:id>/delete', views.delete, name = 'delete'),
]
if settings.DEBUG == True:
    urlpatterns +=static(settings.MEDIA_URL,document_root =
settings.MEDIA_ROOT)
```

Settings.py:

```
import os
from pathlib import Path

# Build paths inside the project like this: BASE_DIR / 'subdir'.
BASE_DIR = Path(__file__).resolve().parent.parent

# SECURITY WARNING: keep the secret key used in production secret!
SECRET_KEY = 'jklbsf_-@jf9viuful+b_vjo@k+nz8yr+k*psexw(&!u3z1y59'

# SECURITY WARNING: don't run with debug turned on in production!
DEBUG = True

ALLOWED_HOSTS = []

# Application definition

INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    recipeapp.apps.RecipeappConfig',
]

MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]

ROOT_URLCONF = 'RecipeProject.urls'

TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [BASE_DIR / 'templates']
    },
    {
        'APP_DIRS': True,
```

```

        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.contrib.auth.context_processors.auth',
                'django.contrib.messages.context_processors.messages',
            ],
        },
    },
]

```

```
WSGI_APPLICATION = 'RecipeProject.wsgi.application'
```

```

# Database
# https://docs.djangoproject.com/en/3.1/ref/settings/#databases

```

```

DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.sqlite3',
        'NAME': BASE_DIR / 'db.sqlite3',
    }
}

```

```

# Password validation
#
https://docs.djangoproject.com/en/3.1/ref/settings/#auth-password-validators

```

```

AUTH_PASSWORD_VALIDATORS = [
    {
        'NAME':
        'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',
    },
    {
        'NAME':
        'django.contrib.auth.password_validation.MinimumLengthValidator',
    },
    {
        'NAME':
        'django.contrib.auth.password_validation.CommonPasswordValidator',
    },
]

```

```
        'NAME':  
'django.contrib.auth.password_validation.NumericPasswordValidator',  
    },  
]
```

```
# Internationalization  
# https://docs.djangoproject.com/en/3.1/topics/i18n/
```

```
LANGUAGE_CODE = 'en-us'
```

```
TIME_ZONE = 'UTC'
```

```
USE_I18N = True
```

```
USE_L10N = True
```

```
USE_TZ = True
```

```
# Static files (CSS, JavaScript, Images)  
# https://docs.djangoproject.com/en/3.1/howto/static-files/
```

```
STATIC_URL = '/static/'
```

```
STATICFILES_DIRS = (os.path.join(BASE_DIR, 'static'), )
```

```
STATIC_ROOT = os.path.join(BASE_DIR, 'staticfiles')
```

```
MEDIA_URL = '/media/'
```

```
MEDIA_ROOT = os.path.join(BASE_DIR, 'media')
```