

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación
Finalidad del Curso
Contenido del Curso

Django

Introducción
Configurar ambiente
de trabajo
Crear proyecto
Crear aplicaciones
Modelos
Relaciones
Django shell y
Querysets
Configurar URLs y
primera views
Sistema de plantillas
Configurar estáticos
Listar Registros
Anexar registros
Modificar y eliminar
registros
Gestion de mensajes
View's
Crear Login
Registro de Usuarios
Login required
Recuperar contraseña
por correo

Framework Django 1.11

nivel: inicial

Instructor Ing. Erwin Paredes

CENDITEL

19 de septiembre de 2017

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y

Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

Fundación CENDITEL

Somos el Centro Nacional de Desarrollo e Investigación en Tecnologías Libres, cuyas actividades son el desarrollo de proyectos con impacto tecnológico enmarcados en las áreas de Software, Hardware y Telecomunicaciones.

<https://www.cenditel.gob.ve>

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestion de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

Finalidad

Consolidar la comunidad de desarrollo en software libre alrededor de la tecnología, en este caso el framework de desarrollo Django del lenguaje de programación Python

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

Contenido del Curso

- Introducción
- Configurar ambiente de trabajo
- Crear proyecto
- Crear aplicaciones y estructura del proyecto
- Modelos de datos y migraciones
- Relaciones entre tablas
- Django shell y Querysets
- Configurar URLs y primera views
- Sistema de plantillas
- Configurar archivos estáticos
- Listar Registros
- Formularios y vista para anexar registros
- Vistas para modificar y eliminar registros

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestion de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

Contenido del Curso

- Gestion de mensajes
- ListView, CreateView, UpdateView, DeleteView
- Crear Login
- Registro de Usuarios
- Decorador Login required
- Recuperar contraseña por correo
- Introducción a conceptos avanzados

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

Django: Es un entorno de trabajo para el desarrollo Web, basado en el lenguaje de programación Python.

Un entorno de trabajo o framework, implica una estructura conceptual y conjunto de herramientas que típicamente son un conjunto de librerías, que permite organizar el desarrollo.

Para el framework Django la estructura conceptual es el denominado MVC (Modelo Vista Controlador), esto es, se organiza el desarrollo en módulos que separan el modelo de datos, la interfaz o vista y el comportamiento.

Por lo que para cada aplicación o módulo que conforma el proyecto se agrupa en lo siguientes archivos:

- `models.py` :Contiene las clases que definen la estructura de datos
- `views.py` :Contiene la codificación de las funcionalidades que permiten controlar el comportamiento del sistema.
- `*.html` :Contiene la codificación de la interfaz

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación
Finalidad del Curso
Contenido del Curso

Django

Introducción
**Configurar ambiente
de trabajo**
Crear proyecto
Crear aplicaciones
Modelos
Relaciones
Django shell y
Querysets
Configurar URLs y
primera views
Sistema de plantillas
Configurar estáticos
Listar Registros
Anexar registros
Modificar y eliminar
registros
Gestion de mensajes
View's
Crear Login
Registro de Usuarios
Login required
Recuperar contraseña
por correo

Sitio Oficial

<https://www.djangoproject.com/>

Instalación

<https://docs.djangoproject.com/en/1.11/intro/install>

Descarga:

<https://www.djangoproject.com/download/>

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

Instalación

```
// Descomprimir el archivo
```

```
tar -xzvf Django1.11.tar.gz
```

```
cd Django1.11
```

```
python setup.py install
```

```
// Comprobar la instalación del framework:
```

```
python  
import django  
django.VERSION
```

```
Ctrl + D
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

```
// Crear el primer proyecto:
```

```
django-admin startproject curso
```

```
// este comando crea la siguiente infraestructura  
de carpetas y archivos:
```

```
curso/  
  manage.py  
  curso/  
    __init__.py  
    settings.py  
    urls.py  
    wsgi.py
```

Configurar la base de datos

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

```
// Se debe haber instalado el gestor de la Base de
Datos Postgresql y la librería de conexión psycopg2
```

```
// Creamos la base de datos:
```

```
su postgres
```

```
psql
```

```
CREATE USER administrador;
```

```
CREATE DATABASE curso;
```

```
// Se edita el archivo curso/settings.py la constante
```

```
DATABASES = {
```

```
    'default': {
```

```
        'ENGINE': 'django.db.backends.postgresql',
```

```
        'NAME': 'curso',
```

```
        'USER': 'administrador',
```

```
        'PASSWORD': 'clave',
```

```
        'HOST': '127.0.0.1',
```

```
        'PORT': '5432',
```

```
    } }
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

```
// Se guarda y ejecuta el siguiente comando para  
crear las tablas iniciales del proyecto:
```

```
python manage.py migrate
```

```
// Se ejecuta el comando para crear el superusuario o  
usuario administrador del proyecto:
```

```
python manage.py createsuperuser
```

```
// Se introducen los datos que solicita
```

```
// Se ejecuta el siguiente comando para activar el  
servidor de prueba de forma local
```

```
python manage.py runserver
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación
Finalidad del Curso
Contenido del Curso

Django

Introducción
Configurar ambiente
de trabajo
Crear proyecto
Crear aplicaciones
Modelos
Relaciones
Django shell y
Querysets
Configurar URLs y
primera views
Sistema de plantillas
Configurar estáticos
Listar Registros
Anexar registros
Modificar y eliminar
registros
Gestión de mensajes
View's
Crear Login
Registro de Usuarios
Login required
Recuperar contraseña
por correo

```
python manage.py runserver
```

```
// Arroja el siguiente mensaje  
Performing system checks...
```

```
System check identified no issues (0 silenced).
```

```
July 26, 2017 - 15:32:30
```

```
Django version 1.11.3, using settings 'curso.settings'  
Starting development server at http://127.0.0.1:8000/  
Quit the server with CONTROL-C.
```

```
// Abrimos el navegador de preferencia y colocamos la  
siguiente dirección para acceder al sistema  
administrativo del proyecto:
```

```
http://localhost:8000/admin
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y

Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

```
http://localhost:8000/admin
```

```
// En el debemos autenticarnos con el usuario y  
password del superusuario creado en los pasos  
anteriores
```

```
// Allí podemos gestionar los usuarios del sistema  
así como los grupos que eventualmente se utilizan  
para asignar permisos de acceso.
```

```
// En el terminal presionar las teclas Ctrl + C para  
interrumpir la ejecución del servidor y así crear  
las aplicaciones del proyecto.
```

```
Modificar el archivo curso/settings.py
```

```
LANGUAGE_CODE = 'es-ve'
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación
Finalidad del Curso
Contenido del Curso

Django

Introducción
Configurar ambiente
de trabajo
Crear proyecto
Crear aplicaciones

Modelos

Relaciones
Django shell y
Querysets
Configurar URLs y
primera views
Sistema de plantillas
Configurar estáticos
Listar Registros
Anexar registros
Modificar y eliminar
registros
Gestión de mensajes
View's
Crear Login
Registro de Usuarios
Login required
Recuperar contraseña
por correo

```
python manage.py startapp encuesta
```

```
// Se edita el archivo encuesta/models.py
```

```
from django.db import models
```

```
class Question(models.Model):
```

```
    question_text = models.CharField(max_length=200)
```

```
    pub_date =models.DateTimeField('date published')
```

```
class Choice(models.Model):
```

```
    question = models.ForeignKey(Question,
```

```
        on_delete=models.CASCADE)
```

```
    choice_text = models.CharField(max_length=200)
```

```
    votes = models.IntegerField(default=0)
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

```
// Almacenamos y editamos el archivo  
curso/settings.py la constante:
```

```
INSTALLED_APPS = [  
    ...  
    'django.contrib.staticfiles',  
    'encuesta',  
]
```

```
// Para crear las tablas correspondiente de models.py  
se ejecuta el comando:
```

```
python manage.py makemigrations encuesta
```

```
python manage.py migrate encuesta
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

```
// Para el manejo de las tablas por medio del  
framework se puede realizar desde el terminal  
con el comando:
```

```
python manage.py shell
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

Views

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

```
// Allí podemos ejecutar las siguientes instrucciones
```

```
from encuesta.models import Question, Choice
from django.utils import timezone
```

```
Question.objects.all()
```

```
q = Question(question_text="Que hay de nuevo?",
              pub_date=timezone.now())
```

```
q.save()
```

```
q.id
```

```
q.question_text
```

```
q.pub_date
```

```
q.question_text = "Q ai d nuevo?"
```

```
q.save()
```

```
Ctrl + D para salir
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

Editamos nuevamente el archivo encuesta/models.py para añadir el siguiente método a la clase Question

```
import datetime

from django.db import models
from django.utils import timezone

class Question(models.Model):
    # ...
    def was_published_recently(self):
        return self.pub_date >= timezone.now() -
            datetime.timedelta(days=1)
```

Guardamos y volvemos a terminal:

```
python manage.py shell
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

```
from polls.models import Question, Choice
```

```
Question.objects.all()
```

```
q = Question.objects.get(pk=1)
```

```
q.was_published_recently()
```

```
q = Question.objects.filter(id=1)
```

```
q = Question.objects.filter(  
    question_text__startswith='Q')
```

```
from django.utils import timezone
```

```
current_year = timezone.now().year
```

```
Question.objects.filter(pub_date__year=current_year)
```

```
Question.objects.get(id=3) Error...
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

```
q = Question.objects.get(pk=1)
q.choice_set.all()
q.choice_set.create(choice_text='No mucho', votes=0)
q.choice_set.create(choice_text='Muchas cosas'
                    ,votes=0)
c = q.choice_set.create(choice_text='De todo un poco'
                    ,votes=0)
```

```
c.question
```

```
q.choice_set.all()
q.choice_set.count()
```

Ctrl + D para salir

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

Incluir el modelo de la aplicación encuesta
en el sistema admin:

```
Editamos el archivo encuesta/admin.py  
from django.contrib import admin
```

```
from .models import Question
```

```
admin.site.register(Question)
```

Guardamos y ejecutamos el servidor
de prueba nuevamente:

```
python manage.py runserver
```

en el navegador <http://localhost:8000/admin>

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

**Configurar URLs y
primera views**

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

Views

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

Crear la primera vista para acceso del proyecto
Editamos el archivo encuesta/urls.py

```
from django.conf.urls import url
from . import views
urlpatterns = [
    # ex: /encueta/
    url(r'^$', views.index, name='index'),
    # ex: /encuesta/5/
    url(r'^(?P<question_id>[0-9]+)/$',
        views.detail, name='detail'),
    # ex: /encuesta/5/results/
    url(r'^(?P<question_id>[0-9]+)/results/$',
        views.results, name='results'),
    # ex: /encuesta/5/vote/
    url(r'^(?P<question_id>[0-9]+)/vote/$',
        views.vote, name='vote'),
]
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y

Querysets

**Configurar URLs y
primera views**

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

Views

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

Editamos el archivo encuesta/views.py:

```
from django.http import HttpResponse
from .models import Question
```

```
def index(request):
    question_list = Question.objects.order_by('-pub_date')[:5]
    output = ', '.join([q.question_text for q in question_list])
    return HttpResponse(output)
```

```
def detail(request, question_id):
    return HttpResponse("Question No: %s." % question_id)
```

```
def results(request, question_id):
    response = "Results of question %s."
    return HttpResponse(response % question_id)
```

```
def vote(request, question_id):
    return HttpResponse("Voting on question %s." % question_id)
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y

Querysets

**Configurar URLs y
primera views**

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

Views

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

Ediatmos el archivo `curso/urls.py` y anexamos:

```
from django.conf.urls import url,include
```

```
urlpatterns = [  
    url(r'^admin/', admin.site.urls),  
    url(r'^encuesta/', include('encuesta.urls')),  
]
```

Guardamos y ejecutamos el servidor de prueba y probamos los enlaces desde el navegador:

```
python manage.py runserver
```

En el navegador probamos los siguientes enlaces:

```
http://localhost:8000/encueta
```

```
http://localhost:8000/encueta/1
```

```
http://localhost:8000/encueta/1/results
```

```
http://localhost:8000/encueta/1/vote
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación
Finalidad del Curso
Contenido del Curso

Django

Introducción
Configurar ambiente
de trabajo
Crear proyecto
Crear aplicaciones
Modelos
Relaciones
Django shell y
Querysets
Configurar URLs y
primera views
Sistema de plantillas
Configurar estáticos
Listar Registros
Anexar registros
Modificar y eliminar
registros
Gestion de mensajes
View's
Crear Login
Registro de Usuarios
Login required
Recuperar contraseña
por correo

Uso del plantillas:

Creamos la siguiente estructura

```
cd encuesta
mkdir templates
mkdir templates/encuesta
```

y editamos el archivo `index.html` allí

```
gedit encuesta/templates/encuesta/index.html
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y

Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

```
{% if question_list %}
    <ul>
        {% for question in question_list %}
            <li>
                <a href="/encuesta/{{ question.id }}/">
                    {{
                        question.question_text
                    }}
                </a>
            </li>
        {% endfor %}
    </ul>
{% else %}
    <p>No hay encuestas.</p>
{% endif %}
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y

Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

Guardamos y modificamos el archivo encuesta views.py para que utilice la plantilla

```
from django.http import HttpResponse
from django.template import loader
```

```
from .models import Question
```

```
def index(request):
    question_list = Question.objects.order_by('-pub_date')[:5]
    template = loader.get_template('encuesta/index.html')
    context = {
        'question_list': question_list,
    }
    return HttpResponse(template.render(context, request))
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

o utilizando la shortcuts render

```
from django.shortcuts import render
```

```
from .models import Question
```

```
def index(request):  
    question_list = Question.objects.order_by('-pub_date')[:5]  
    context = {'question_list': question_list}  
    return render(request, 'encuesta/index.html', context)
```

Descargar plantillas prediseñadas, por ejemplo:

<https://adminlte.io/>

Copiar los directorios ccs y js en los en la carpeta static

Copiar los archivo *.html en la carpeta templates

Editar index.html y anexar en las primeras lineas:

```
{% load static %}
```

Cambiar los enlaces a cada archivo en:

```
<link rel="stylesheet"  
href="{% static 'css/DataTables/jquery.dataTables.min.css' %}">
```

Y anexar la siguiente linea al settings.py

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

Creamos el archivo ajax.py en el directorio curso/curso

```
# -*- encoding: utf-8 -*-
from django.conf import settings
from django_datatables_view.base_datatable_view import (
    BaseDatatableView)

from django.contrib.auth.models import (
    User)

class ListUsersAjaxView(BaseDatatableView):
    model = User
    columns = ['pk', 'first_name', 'last_name', 'username', 'email',
               'date_joined', 'last_joined' ]
    order_columns = ['pk', 'username']
    max_display_length = 500

    def __init__(self):
        super(ListUsersAjaxView, self).__init__()

    def get_initial_queryset(self):
        return self.model.objects.all()
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

```
def prepare_results(self, qs):
    json_data = []
    for item in qs:
        json_data.append([
            item.username,
            "{0} {1}".format(str(item.first_name),str(item.last_name)),
            item.email,
            item.date_joined.strftime("%Y-%m-%d %H:%M:%S"),
            item.last_login
        ])

    return json_data

# Fin del archivo ajax.py

#En el archivo urls.py anexar las siguientes lineas
from .ajax import *

urlpatterns = [
    ....
    url(r'^listar-users/$', ListUsersAjaxView.as_view(),
        name="listar_users"),
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

Views

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

Se anexa el siguiente código el archivo index.html
En la parte visual:

```
<div id="datatable"></div>
```

Y en la parte de código javascript

```
<script type="text/javascript">
$(document).ready(function() {
    $('#datatable').dataTable({
        "processing": true,
        "serverSide": true,
        "ajax": {% url 'listar_user'%},
        language: {url: JSON_DATA}
    });
    $('#datatable')
        .removeClass('display')
        .addClass('table table-striped table-bordered');
});
</script>
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

Creamos el método en el archivo views.py

```
def AnexarRegistro(request):
    if request.method == 'POST':

        vusername = request.POST['username']
        u = User(username = vusername)
        u.save()
        message = _("El usuario fue creado")
        template = loader.get_template('personal/profile.html')
        context = {'message':message}
        return HttpResponse(template.render(context, request))
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

**Modificar y eliminar
registros**

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

```
@login_required
def change_profile(request):
    """ Cambiar perfil de usuario
    """
    if request.method == 'POST':
        user_id = request.POST['user_id']
        personal = User.objects.get(user_id=request.user.id)
        personal.first_name = request.POST['fname']
        personal.last_name = request.POST['lname']
        personal.email = request.POST['email']
        personal.is_active = request.POST.get('is_active')
        personal.user.save()

        message = _("The profile change was made")
    context = {'personal':personal, 'message':message,}
    template = loader.get_template('personal/profile.html')
    return HttpResponse(template.render(context, request))
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

**Modificar y eliminar
registros**

Gestion de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

```
@login_required
def delete_personal(request):
    """ Delete users
    """
    id = json.loads(request.POST['seleccionados'])
    print id

    for id_values in id:
        if id_values["id"] != "0":

            u = User.objects.get(pk=int(id_values["id"]))
            u.delete()

    return redirect('personal')
```

En la vista se une el mensaje al request

```
from django.contrib import messages
```

```
@login_required
```

```
def change_profile(request):
```

```
    """ Cambiar perfil de usuario
```

```
    """
```

```
    if request.method == 'POST':
```

```
        user_id = request.POST['user_id']
```

```
        personal = User.objects.get(user_id=request.user.id)
```

```
        personal.first_name = request.POST['fname']
```

```
        personal.last_name = request.POST['lname']
```

```
        personal.email = request.POST['email']
```

```
        personal.is_active = request.POST.get('is_active')
```

```
        personal.save()
```

```
        messages.add_message(request, messages.INFO,
```

```
                               'Fue modificado')
```

```
    return redirect('member')
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestion de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

En el template

```
{% if messages %}
<ul class="messages">
  {% for message in messages %}
    <li{% if message.tags %} class="{ { message.tags } }"
      {% endif %}>
      {{ message }}
    </li>
  {% endfor %}
</ul>
{% endif %}
```

Así podemos utilizar los diversos niveles:

```
messages.debug(request, '%s Cuantos.' % count)
messages.info(request, 'Informacion importante.')
messages.success(request, 'Fue modificado.')
messages.warning(request, 'Advertencia.')
messages.error(request, 'Error 404.')
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

```
#### ListView
```

```
# En el archivo views.py
```

```
from django.views.generic.list import ListView
from django.utils import timezone
```

```
from articles.models import Article
```

```
class ArticleListView(ListView):
```

```
    model = Article
```

```
    def get_context_data(self, **kwargs):
        context = super(ArticleListView,
                        self).get_context_data(**kwargs)
        context['now'] = timezone.now()
        return context
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y

Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

```
# En el archivo urls.py
from django.conf.urls import url

from article.views import ArticleListView

urlpatterns = [
    url(r'^$', ArticleListView.as_view(), name='article-list'),
]

# El template
<h1>Articles</h1>
<ul>
{% for article in object_list %}
    <li>
        {{ article.pub_date|date }} - {{ article.headline }}
    </li>
{% empty %}
    <li>No articles yet.</li>
{% endfor %}
</ul>
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

```
#### DetailView
```

```
# En el archivo views.py
```

```
from django.views.generic.detail import DetailView
from django.utils import timezone
```

```
from articles.models import Article
```

```
class ArticleDetailView(DetailView):
```

```
    model = Article
```

```
    def get_context_data(self, **kwargs):
        context = super(ArticleDetailView, self).
            get_context_data(**kwargs)
        context['now'] = timezone.now()
        return context
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación
Finalidad del Curso
Contenido del Curso

Django

Introducción
Configurar ambiente
de trabajo
Crear proyecto
Crear aplicaciones
Modelos
Relaciones
Django shell y
Querysets
Configurar URLs y
primera views
Sistema de plantillas
Configurar estáticos
Listar Registros
Anexar registros
Modificar y eliminar
registros
Gestión de mensajes
View's
Crear Login
Registro de Usuarios
Login required
Recuperar contraseña
por correo

```
# En el archivo urls.py
```

```
from django.conf.urls import url
```

```
from article.views import ArticleDetailView
```

```
urlpatterns = [  
    url(r'^(?P<slug>[-\w]+)/$', ArticleDetailView.as_view(),  
        name='article-detail'),  
]
```

```
# El template
```

```
<h1>{{ object.headline }}</h1>  
<p>{{ object.content }}</p>  
<p>Reporter: {{ object.reporter }}</p>  
<p>Published: {{ object.pub_date|date }}</p>  
<p>Date: {{ now|date }}</p>
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación
Finalidad del Curso
Contenido del Curso

Django

Introducción
Configurar ambiente
de trabajo
Crear proyecto
Crear aplicaciones
Modelos
Relaciones
Django shell y
Querysets
Configurar URLs y
primera views
Sistema de plantillas
Configurar estáticos
Listar Registros
Anexar registros
Modificar y eliminar
registros
Gestión de mensajes
View's
Crear Login
Registro de Usuarios
Login required
Recuperar contraseña
por correo

```
#### CreateView
```

```
from django.views.generic.edit import CreateView  
from myapp.models import Author
```

```
class AuthorCreate(CreateView):  
    model = Author  
    fields = ['name']
```

```
# En el template
```

```
<form action="" method="post">{% csrf_token %}  
    {{ form.as_p }}  
    <input type="submit" value="Save" />  
</form>
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación
Finalidad del Curso
Contenido del Curso

Django

Introducción
Configurar ambiente
de trabajo
Crear proyecto
Crear aplicaciones
Modelos
Relaciones
Django shell y
Querysets
Configurar URLs y
primera views
Sistema de plantillas
Configurar estáticos
Listar Registros
Anexar registros
Modificar y eliminar
registros
Gestión de mensajes
View's
Crear Login
Registro de Usuarios
Login required
Recuperar contraseña
por correo

```
#### UpdateView
```

```
from django.views.generic.edit import UpdateView  
from myapp.models import Author
```

```
class AuthorUpdate(UpdateView):  
    model = Author  
    fields = ['name']  
    template_name_suffix = '_update_form'
```

```
# En el template
```

```
<form action="" method="post">{% csrf_token %}  
    {{ form.as_p }}  
    <input type="submit" value="Update" />  
</form>
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación
Finalidad del Curso
Contenido del Curso

Django

Introducción
Configurar ambiente
de trabajo
Crear proyecto
Crear aplicaciones
Modelos
Relaciones
Django shell y
Querysets
Configurar URLs y
primera views
Sistema de plantillas
Configurar estáticos
Listar Registros
Anexar registros
Modificar y eliminar
registros
Gestión de mensajes
View's
Crear Login
Registro de Usuarios
Login required
Recuperar contraseña
por correo

```
#### DeleteView
```

```
from django.views.generic.edit import DeleteView  
from django.urls import reverse_lazy  
from myapp.models import Author
```

```
class AuthorDelete(DeleteView):  
    model = Author  
    success_url = reverse_lazy('author-list')
```

```
# En el template
```

```
<form action="" method="post">{% csrf_token %}  
    <p>Are you sure you want to delete "{{ object }}"?</p>  
    <input type="submit" value="Confirm" />  
</form>
```

Creamos la aplicacion user desde la consola:

```
python manage.py startapp user
```

Editamos el archivo user/views.py:

```
# -*- coding: utf-8 -*-

from django.shortcuts import render
from django.contrib import messages
from django.contrib.auth import (
    authenticate, logout, login
)
from django.contrib.auth.models import (
    Group, Permission, User
)

class LoginView(FormView):
    form_class = FormularioLogin
    template_name = 'users.login.html'
    success_url = '/inicio/'
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

```
def form_valid(self, form):

    usuario = form.cleaned_data['usuario']
    contraseña = form.cleaned_data['contraseña']
    usuario = authenticate(username=usuario,password=contraseña)

    if usuario is not None:
        login(self.request, usuario)
        messages.info(self.request, 'Bienvenido %s has ingresado\
        Sistema con el usuario %s \
        ' % (usuario.first_name,
            usuario.username))
    else:
        self.success_url = reverse_lazy('users:login')
        messages.warning(self.request, 'Verifique su nombre y \
        contraseña\
        y vuelve a intentar')

    return super(LoginView, self).form_valid(form)
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

```
#### Crear Permisos desde el shell ejecutar
```

```
from myapp.models import BlogPost
from django.contrib.auth.models import Permission
from django.contrib.contenttypes.models import ContentType

content_type = ContentType.objects.get_for_model(BlogPost)
permission = Permission.objects.create(
    codename='can_publish',
    name='Can Publish Posts',
    content_type=content_type,
)
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

```
from django.contrib.auth.models import Permission, User
from django.contrib.contenttypes.models import ContentType
from django.shortcuts import get_object_or_404
```

```
from myapp.models import BlogPost
```

```
def user_gains_perms(request, user_id):
    user = get_object_or_404(User, pk=user_id)
    # any permission check will
    #cache the current set of permissions
    user.has_perm('myapp.change_blogpost')

    content_type=ContentType.objects.get_for_model(BlogPost)
    permission = Permission.objects.get(
        codename='change_blogpost',
        content_type=content_type,
    )
    user.user_permissions.add(permission)
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación
Finalidad del Curso
Contenido del Curso

Django

Introducción
Configurar ambiente
de trabajo
Crear proyecto
Crear aplicaciones
Modelos
Relaciones
Django shell y
Querysets
Configurar URLs y
primera views
Sistema de plantillas
Configurar estáticos
Listar Registros
Anexar registros
Modificar y eliminar
registros
Gestion de mensajes
View's
Crear Login
Registro de Usuarios
Login required
Recuperar contraseña
por correo

```
from django.contrib.auth.decorators import
    login_required, permission_required
from django.views.generic import TemplateView

from .views import VoteView

urlpatterns = [
    url(r'^about/$', login_required(
        TemplateView.as_view(template_name="secret.html"))),
    url(r'^vote/$', permission_required(
        'polls.can_vote')(VoteView.as_view())),
]
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y

Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestión de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

```
def newpassword(request):
    puser = request.POST['puser']
    try:
        user = User.objects.get(username=puser)
        randompass = ''.join([choice(
            '1234567890qwertyuiopasdfghjklzxcvbnm')
            for i in range(10)])
        print randompass
        subject = _('System: New Password')
        message = _('Your password is reset, new password: ')
                    + randompass
        user.email_user("subject", "message")
        user.set_password(randompass)
        user.save()
    except:
        print "error send mail"
        mensaje = _("User not found")
    return redirect('/authenticate/signin')
```

Instructor Ing.
Erwin Paredes

Cenditel

La Fundación

Finalidad del Curso

Contenido del Curso

Django

Introducción

Configurar ambiente
de trabajo

Crear proyecto

Crear aplicaciones

Modelos

Relaciones

Django shell y
Querysets

Configurar URLs y
primera views

Sistema de plantillas

Configurar estáticos

Listar Registros

Anexar registros

Modificar y eliminar
registros

Gestion de mensajes

View's

Crear Login

Registro de Usuarios

Login required

Recuperar contraseña
por correo

Otros conceptos

- * Internacionalización
- * Zonas Horarias
- * Servidor Web
- * Geodjango
- * Django Rest

Entre otros...

CENDITEL