

¡Claro! Aquí tienes la primera parte del proyecto adaptada para un centro de idiomas, manteniendo el mismo formato:

Primera parte

Proyecto: Centro de Idiomas.

Lenguaje: Python.

Framework: Django.

Editor: VS code.

1 Procedimiento para crear carpeta del Proyecto: UIII_Centro_de_Idiomas_0498

2 Procedimiento para abrir VS Code sobre la carpeta UIII_Centro_de_Idiomas_0498

3 Procedimiento para abrir terminal en VS Code

4 Procedimiento para crear carpeta entorno virtual “.venv” desde terminal de VS Code

5 Procedimiento para activar el entorno virtual.

6 Procedimiento para activar intérprete de Python.

7 Procedimiento para instalar Django

8 Procedimiento para crear proyecto backend_Centro_Idiomas sin duplicar carpeta.

9 Procedimiento para ejecutar servidor en el puerto 8498

10 Procedimiento para copiar y pegar el link en el navegador.

11 Procedimiento para crear aplicación app_Centro_Idiomas

12 Aquí los modelos models.py

==--==--==--

from django.db import models

=====

=====

MODELO: IDIOMA

=====

=====

class Idioma(models.Model):

nombre = models.CharField(max_length=50, unique=True)

nivel = models.CharField(max_length=50)

region = models.CharField(max_length=100, blank=True, null=True)

descripcion = models.TextField(blank=True, null=True)

codigo = models.CharField(max_length=10, unique=True)

bandera_img = models.ImageField(upload_to='banderas/', blank=True, null=True)

```
def __str__(self):  
    return self.nombre
```

=====

=====

MODELO: PROFESOR

=====

=====

```
class Profesor(models.Model):  
    nombre = models.CharField(max_length=100)  
    ap_paterno = models.CharField(max_length=100)  
    ap_materno = models.CharField(max_length=100, blank=True, null=True)  
    telefono = models.CharField(max_length=15, blank=True, null=True)  
    correo = models.EmailField(unique=True)  
    id_idioma = models.ForeignKey(Idioma, on_delete=models.CASCADE,  
    related_name="profesores")
```

```
def __str__(self):  
    return f"{self.nombre} {self.ap_paterno}"
```

=====

=====

MODELO: CLASE

=====

=====

```

class Clase(models.Model):
    nombre = models.CharField(max_length=150)
    descripcion = models.TextField(blank=True, null=True)
    capacidad = models.PositiveIntegerField()
    nivel = models.CharField(max_length=50, choices=[
        ('Básico A1', 'Básico A1'),
        ('Básico A2', 'Básico A2'),
        ('Intermedio B1', 'Intermedio B1'),
        ('Intermedio B2', 'Intermedio B2'),
        ('Avanzado C1', 'Avanzado C1'),
        ('Avanzado C2', 'Avanzado C2')
    ])
    id_profesor = models.ForeignKey(Profesor, on_delete=models.CASCADE,
    related_name="clases")
    idioma = models.ForeignKey(Idioma, on_delete=models.CASCADE,
    related_name="clases_idioma")

    def __str__(self):
        return f"{self.nombre} ({self.idioma.nombre})"

```

==--==--==

12.5 Procedimiento para realizar las migraciones (makemigrations y migrate).

13 Primero trabajamos con el MODELO: IDIOMA

14 En views de app_Centro_Idiomas crear las funciones con sus códigos correspondientes (inicio_centro_idiomas, agregar_idioma, actualizar_idioma, realizar_actualizacion_idioma, borrar_idioma)

15 Crear la carpeta “templates” dentro de “app_Centro_Idiomas”.

16 En la carpeta templates crear los archivos html (base.html, header.html, navbar.html, footer.html, inicio.html).

17 En el archivo base.html agregar bootstrap para CSS y JS.

18 En el archivo navbar.html incluir las opciones (“Sistema de Administración Centro de Idiomas”, “Inicio”, “Idiomas”, en submenú de Idiomas (Agregar Idioma, Ver Idiomas, Actualizar Idioma, Borrar Idioma), “Profesores” en submenú de Profesores (Agregar Profesor, Ver Profesores, Actualizar Profesores, Borrar Profesores)

“Clases” en submenú de Clases (Agregar Clases, Ver Clases, Actualizar Clases, Borrar Clases), incluir iconos a las opciones principales, no en los submenús.

19 En el archivo footer.html incluir derechos de autor, fecha del sistema y “Creado por Ing. Eliseo Nava, Cbtis 128” y mantenerla fija al final de la página.

20 En el archivo inicio.html se usa para colocar información del sistema más una imagen tomada desde la red sobre un centro de idiomas.

21 Crear la subcarpeta idioma dentro de app_Centro_Idiomas/templates.

22 Crear los archivos html con su código correspondientes de (agregar_idioma.html, ver_idiomas.html mostrar en tabla con los botones ver, editar y borrar, actualizar_idioma.html, borrar_idioma.html)

dentro de app_Centro_Idiomas/templates/idioma.

23 No utilizar forms.py.

24 Procedimiento para crear el archivo urls.py en app_Centro_Idiomas con el código correspondiente para acceder a las funciones de views.py para operaciones de CRUD en idiomas.

25 Procedimiento para agregar app_Centro_Idiomas en settings.py de backend_Centro_Idiomas

26 Realizar las configuraciones correspondiente a urls.py de backend_Centro_Idiomas para enlazar con app_Centro_Idiomas

27 Procedimiento para registrar los modelos en admin.py y volver a realizar las migraciones.

27 Por lo pronto solo trabajar con "Idioma" dejar pendiente # MODELO: PROFESOR y # MODELO: CLASE

28 Utilizar colores suaves, atractivos y modernos, el código de las páginas web sencillas.

28 No validar entrada de datos.

29 Al inicio crear la estructura completa de carpetas y archivos.

30 Proyecto totalmente funcional.

31 Finalmente ejecutar servidor en el puerto 8498.