

## Primera parte

**Proyecto:** Preparatoria

**Lenguaje:** Python

**Framework:** Django

**Editor:** VS Code

---

### Procedimientos:

1. Procedimiento para crear la carpeta del proyecto: UIII\_Preparatoria\_0656.
2. Procedimiento para abrir VS Code sobre la carpeta UIII\_Preparatoria\_0656.
3. Procedimiento para abrir la terminal en VS Code.
4. Procedimiento para crear la carpeta del entorno virtual .venv desde la terminal de VS Code.
5. Procedimiento para activar el entorno virtual.
6. Procedimiento para activar el intérprete de Python.
7. Procedimiento para instalar Django.
8. Procedimiento para crear el proyecto backend\_Preparatoria sin duplicar la carpeta.
9. Procedimiento para ejecutar el servidor en el puerto 8656.
10. Procedimiento para copiar y pegar el link en el navegador.
11. Procedimiento para crear la aplicación app\_Preparatoria.
12. Aquí el modelo models.py:

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

```
# =====
```

```
# MODELO: PROFESOR
```

```
# =====
```

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

```
    nombre_profesor = models.CharField(max_length=50, unique=True)
```

```
    apellido_profesor = models.CharField(max_length=50, unique=True)
```

```
    correo_profesor = models.EmailField(max_length=100)
```

```

telefono = models.CharField(max_length=15)
especialidad = models.CharField(max_length=50, default="")
fecha_contratacion = models.DateField(auto_now_add=True)
activo = models.BooleanField(default=True)

def __str__(self):
    return f"{self.nombre_profesor} {self.apellido_profesor}"

# =====
# MODELO: CURSO
# =====

class Curso(models.Model):
    nombre_curso = models.CharField(max_length=50)
    codigo = models.CharField(max_length=10)
    descripcion = models.TextField(max_length=100)
    creditos = models.PositiveIntegerField()
    horario = models.CharField(max_length=50)
    aula = models.CharField(max_length=20)
    profesor = models.ForeignKey(Profesor, related_name="cursos",
on_delete=models.CASCADE)

    def __str__(self):
        return f"{self.nombre_curso} ({self.codigo})"

# =====
# MODELO: ESTUDIANTE
# =====

class Estudiante(models.Model):
    nombre_estudiante = models.CharField(max_length=50)

```

```
apellido_estudiante = models.CharField(max_length=50)
matricula = models.CharField(max_length=10, unique=True)
correo_estudiante = models.EmailField(max_length=100)
fecha_nacimiento = models.DateField()
fecha_inscripcion = models.DateField(auto_now_add=True)
cursos = models.ManyToManyField(Curso, related_name="estudiantes")
```

```
def __str__(self):
    return f"{self.nombre_estudiante} {self.apellido_estudiante}"
```

---

13. Primero trabajamos con el **MODELO: PROFESOR**.
14. En la vista (views.py) de app\_Preparatoria, crear las funciones correspondientes: inicio\_profesor, agregar\_profesor, actualizar\_profesor, realizar\_actualizacion\_profesor, borrar\_profesor.
15. Crear la carpeta **"templates"** dentro de app\_Preparatoria.
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 Preparatoria"**
  - **"Inicio"**
  - **"Profesor"** (en el submenú de Profesor: **"Agregar Profesor"**, **"Ver Profesor"**, **"Actualizar Profesor"**, **"Borrar Profesor"**)
  - **"Curso"** (en el submenú de Curso: **"Agregar Curso"**, **"Ver Curso"**, **"Actualizar Curso"**, **"Borrar Curso"**)
  - **"Estudiante"** (en el submenú de Estudiante: **"Agregar Estudiante"**, **"Ver Estudiante"**, **"Actualizar Estudiante"**, **"Borrar Estudiante"**)
  - Incluir íconos a las opciones principales (no en los submenús).
19. En el archivo footer.html, incluir derechos de autor, fecha del sistema y **"Creado por Aaron Nava, CBTis 128"**, y mantenerla fija al final de la página.

20. En el archivo inicio.html, colocar información del sistema junto con una imagen relacionada con la preparatoria (tomada desde la red).
21. Crear la subcarpeta "**profesor**" dentro de app\_Preparatoria/templates.
22. Crear los archivos HTML correspondientes dentro de app\_Preparatoria/templates/profesor:
  - agregar\_profesor.html
  - ver\_profesor.html (mostrar en tabla con los botones "**Ver**", "**Editar**", "**Borrar**")
  - actualizar\_profesor.html
  - borrar\_profesor.html
23. No utilizar forms.py.
24. Procedimiento para crear el archivo urls.py en app\_Preparatoria con el código correspondiente para acceder a las funciones de views.py para operaciones CRUD en el modelo **Profesor**.
25. Procedimiento para agregar app\_Preparatoria en settings.py de backend\_Preparatoria.
26. Realizar las configuraciones correspondientes en urls.py de backend\_Preparatoria para enlazar con app\_Preparatoria.
27. Procedimiento para registrar los modelos en admin.py y volver a realizar las migraciones.
28. Por el momento, solo trabajar con "**Profesor**" y dejar pendiente el **MODELO: CURSO** y el **MODELO: ESTUDIANTE**.
29. Utilizar colores suaves, atractivos y modernos para las páginas web.
30. No validar la entrada de datos.
31. Al inicio, crear la estructura completa de carpetas y archivos.
32. Asegurarse de que el proyecto esté completamente funcional.
33. Finalmente, ejecutar el servidor en el puerto 8656.