



Módulo 6: Desarrollo de aplicaciones web con Python y Django

AE1: Conocer Django y crear un proyecto desde cero.

Hoy trabajaremos con teoría + práctica en VS Code.



¿Qué es Django?

Framework web en Python.

Rápido, seguro y escalable.

Usado en grandes empresas (Instagram, Spotify, Pinterest).

Ventajas de Django

1

Seguridad integrada
Protección contra
vulnerabilidades comunes

2

Reutilización de código
Componentes modulares y
extensibles

3

Amplia comunidad
Soporte y recursos disponibles

4

Admin automático
Interfaz de administración lista para usar

5

Documentación excelente
Recursos completos para aprendizaje

Django vs otros frameworks

Django

Enfoque "baterías incluidas".

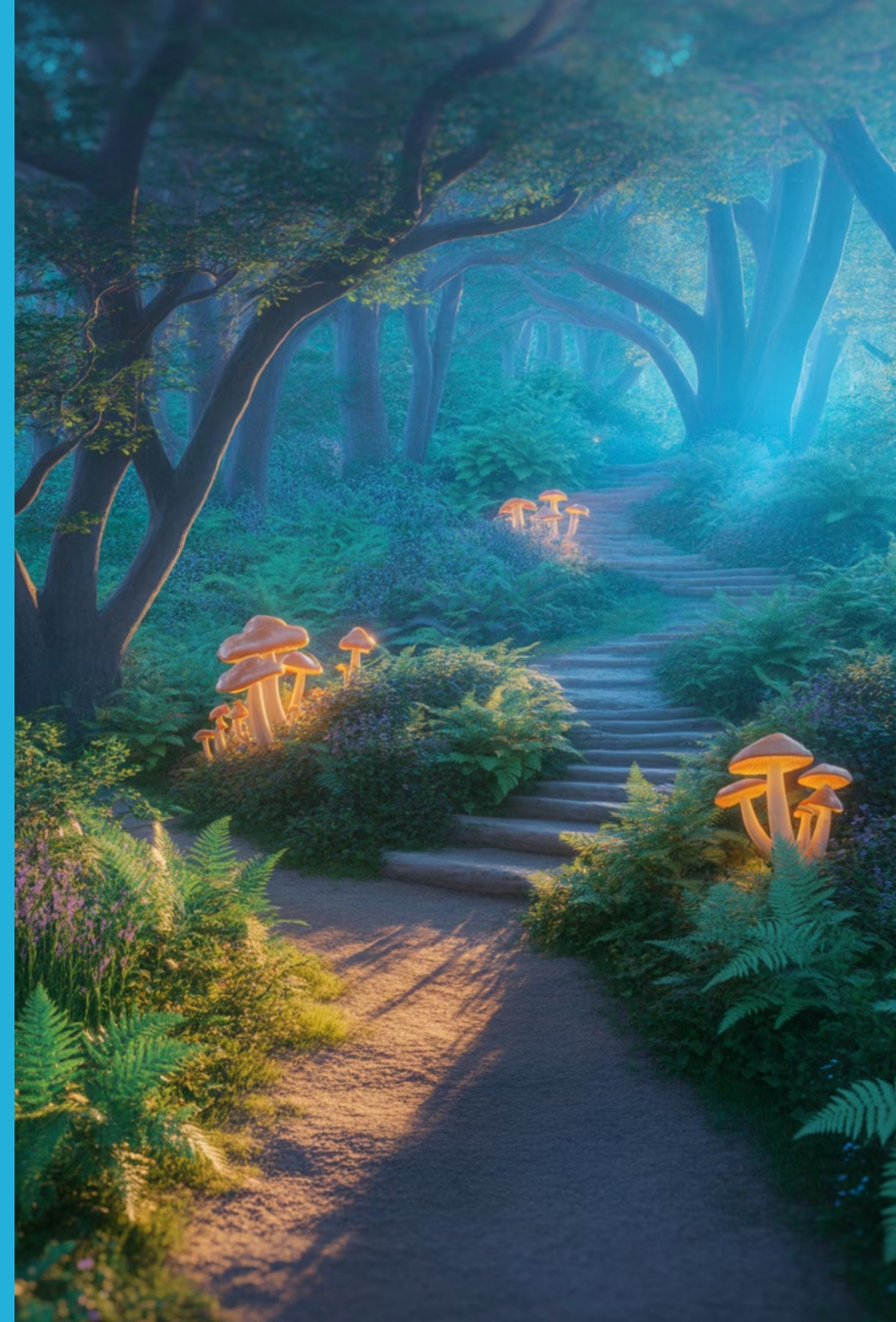
Flask

Minimalista, flexible.

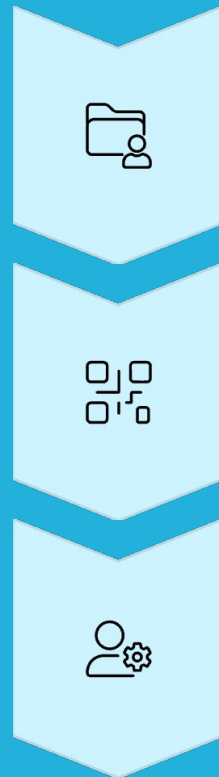
FastAPI

Rendimiento y APIs modernas.

Actividad activa: ¿Qué prefieren? ¿Un framework todo en uno o uno minimalista?



Patrón MTV



Modelo
Datos

Template
Interfaz

Vista
Lógica de negocio

Ejemplo: Un blog → Modelo(Post), Template(HTML), Vista(qué mostrar).

Model-View-Template



Instalando Django

Paso 1

Crear entorno virtual.

Paso 2

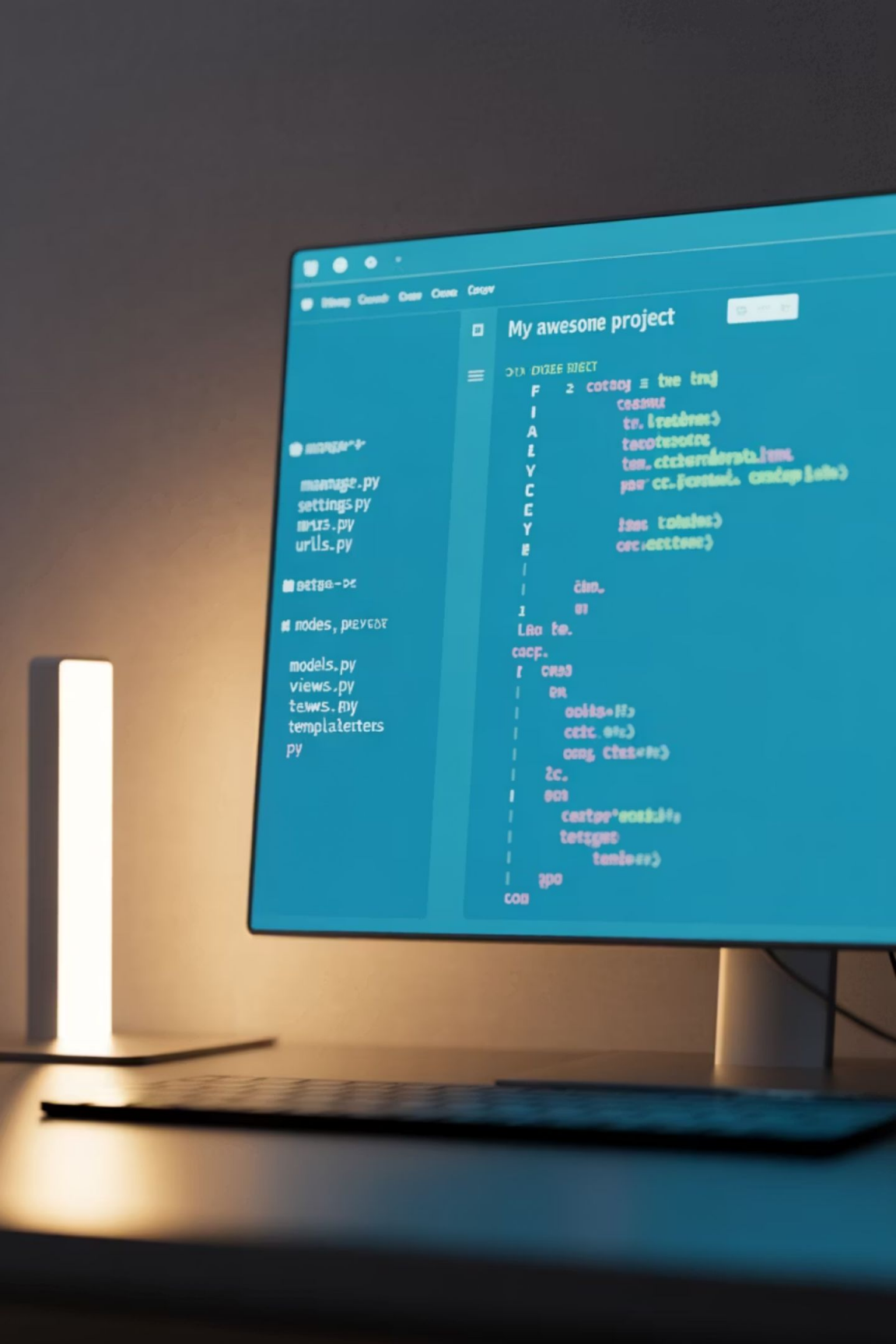
Activarlo en VS Code.

Paso 3

Instalar Django con `pip install django`.

Mini-desafío: Estudiantes prueban instalar Django y mostrar versión.





Crear un proyecto

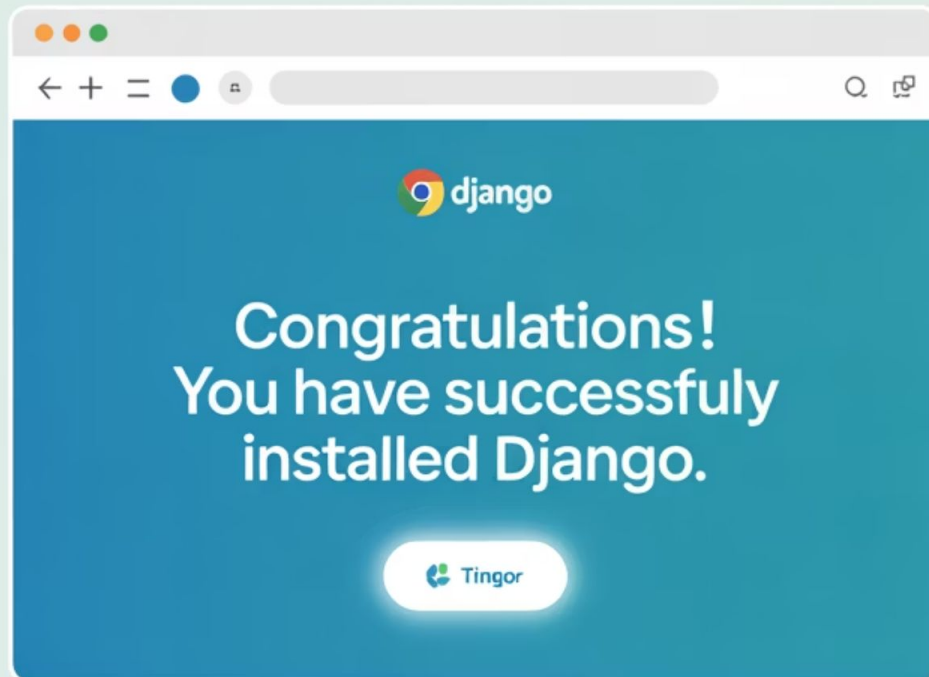
Comando:

```
django-admin startproject mi_proyecto
```

Esto genera:

- `manage.py`
- Carpeta de configuración (`settings`, `urls`, etc.)

Django's installation



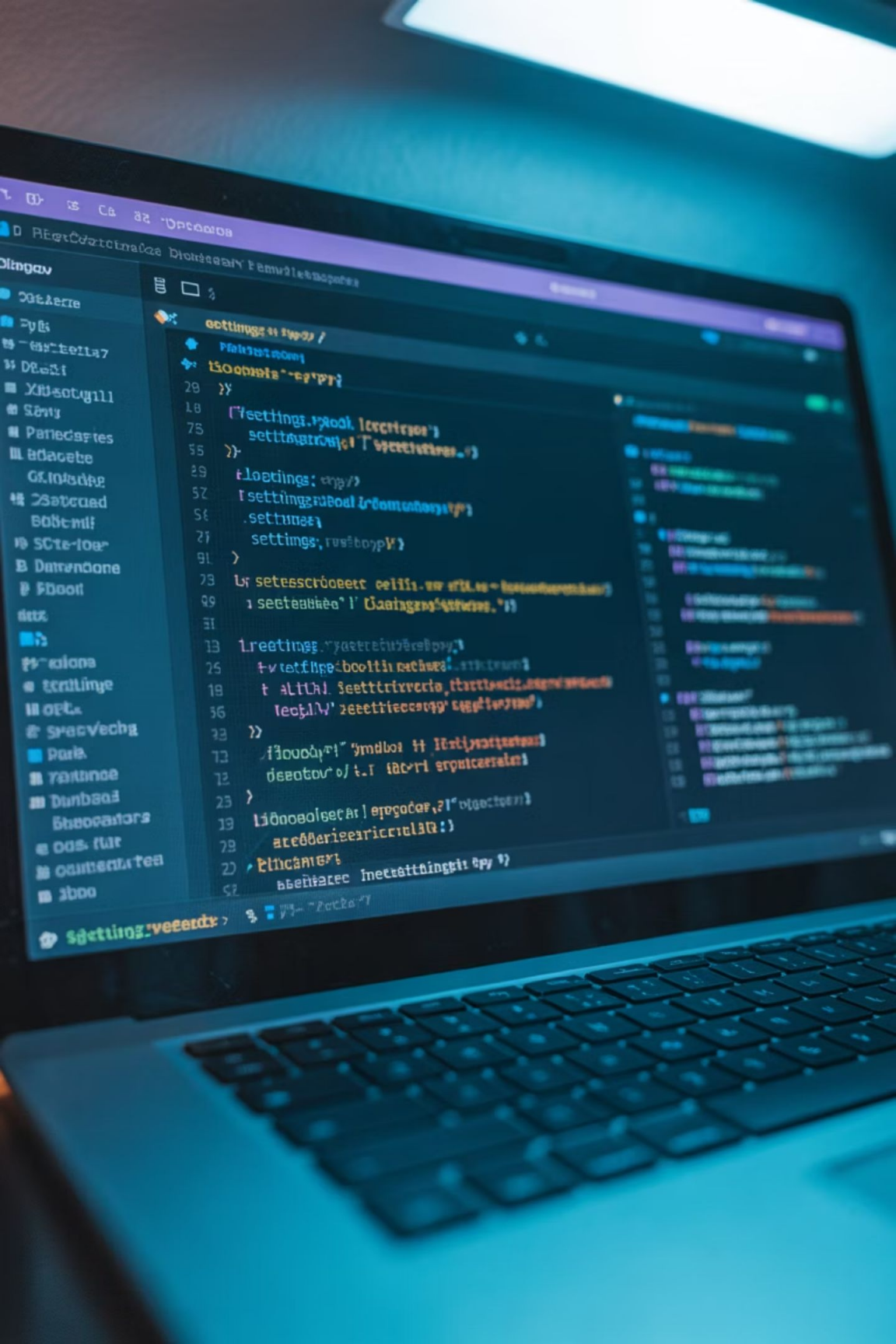
Ejecutar servidor

Comando:

```
python manage.py runserver
```

Abrir en navegador: <http://127.0.0.1:8000>

👉 Ver la pantalla de bienvenida de Django.



Crear una app

Comando:

```
python manage.py startapp blog
```

👉 Una app es un módulo independiente. Ejemplo: blog, usuarios, carrito de compras.



Configuración en settings.py

Código:

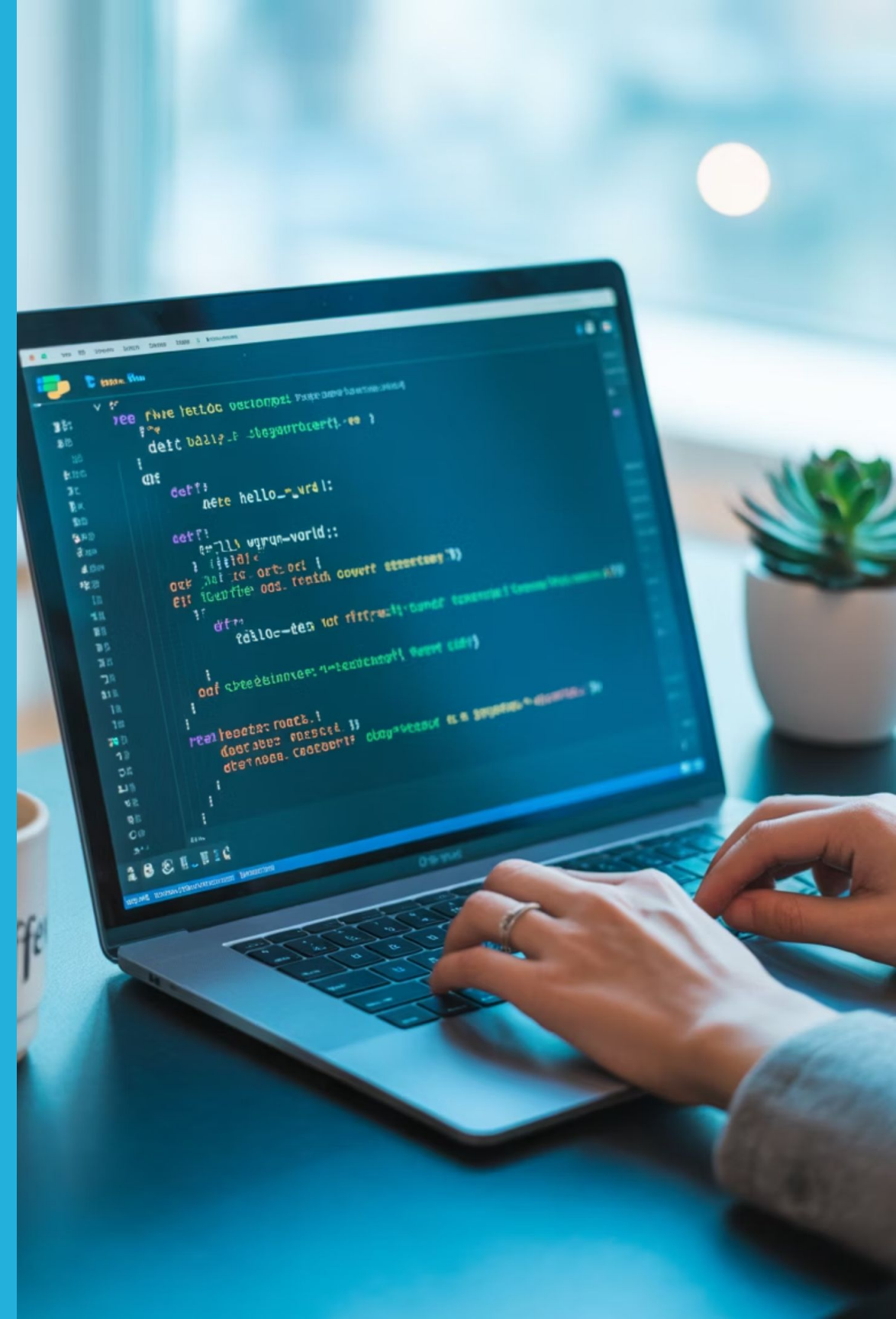
```
INSTALLED_APPS = [    ...,    'blog',]
```

👉 Registramos la nueva app en el proyecto.

Nuestra primera vista

Código en views.py:

```
from django.http import HttpResponseRedirect  
def inicio(request): return  
    HttpResponseRedirect("/")
```

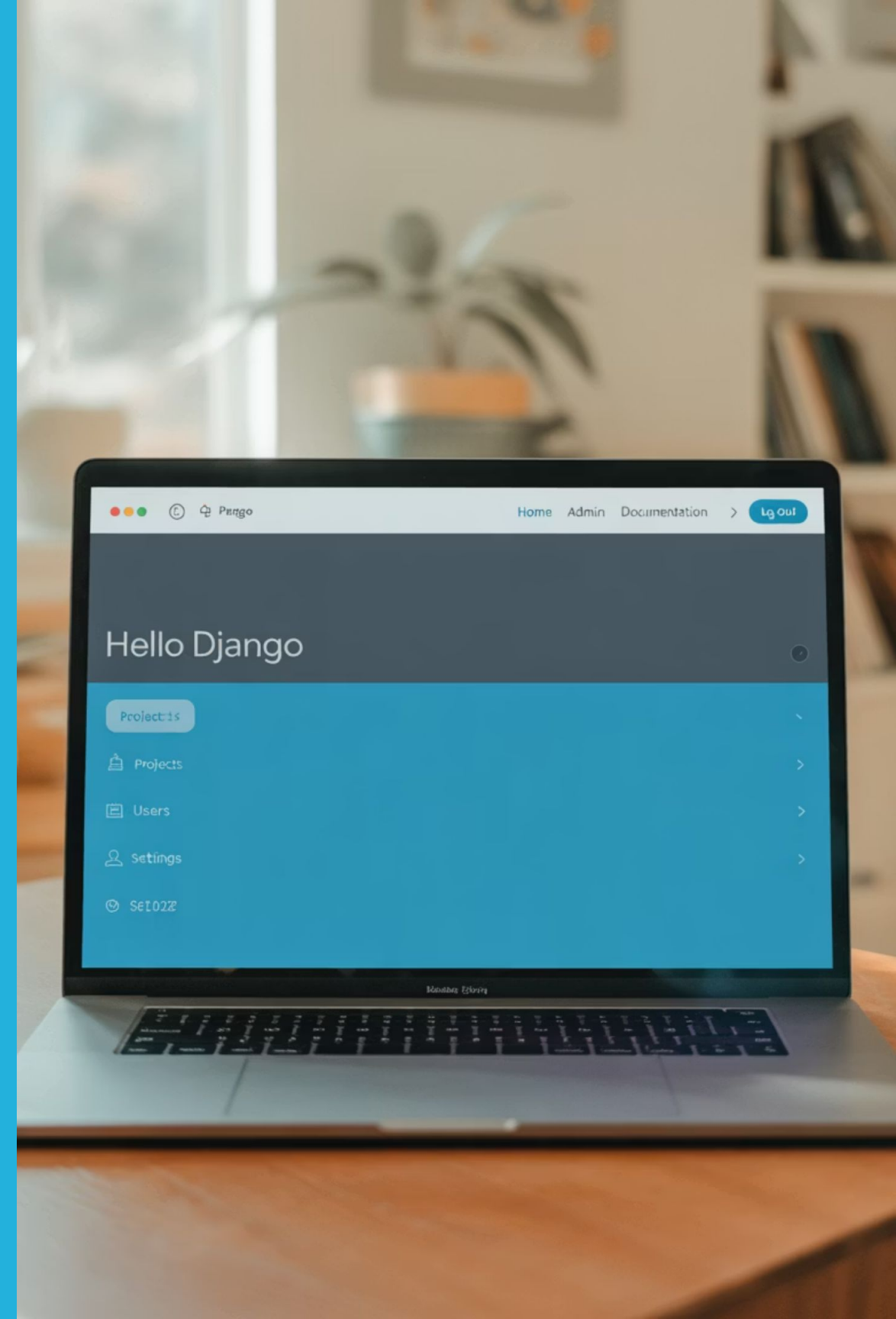


Conectar la vista en urls.py

Código en urls.py:

```
from django.urls import pathfrom blog import viewsurlpatterns = [
    path('', views.inicio),]
```

👉 Ahora la app responde en el navegador.





Actividad práctica

Ejercicio en parejas:



Estudiante A

Crea la vista



Estudiante B

Configura la URL



Cambio

Luego cambian roles

Django Admin

Log in

[Forgot password?](#)

Powered by **Django**

Explorando el Admin de Django

Admin listo para usar.

Se activa con:

```
python manage.py createsuperuser
```

Permite gestionar usuarios, modelos y permisos.

Entorno de desarrollo vs producción

Desarrollo

- Servidor local
- Modo debug activado
- Facilita pruebas

👉 Pregunta activa: ¿Qué pasaría si dejamos DEBUG=True en producción?

Producción

- Seguridad
- Rendimiento
- Sin debug



Debate guiado

👉 Tema: ¿Es más importante comenzar configurando la base de datos o creando vistas?

Grupo A: Base de datos

- Estructura de datos primero
- Modelos definen la aplicación
- Enfoque "datos primero"

Grupo B: Vistas

- Interfaz de usuario primero
- Experiencia del usuario
- Enfoque "frontend primero"





Buenas prácticas iniciales



Entornos virtuales

Aislar dependencias del proyecto



Requirements.txt

Mantener lista de dependencias



Apps pequeñas

Dividir el proyecto en módulos



Control de versiones

Usar Git para seguimiento

Resumen del día



Qué es Django

Framework web en Python



Cómo crear proyecto y app

Estructura básica de aplicación



Configuración en settings

Registro de aplicaciones



Primera vista y URL

Respuesta HTTP básica



Admin de Django

Interfaz de administración

Django



Mini-desafío para casa



Crea una app llamada "tienda" que muestre en la URL principal:



"Bienvenido a mi tienda online con Django".

Sube captura al foro.

Cierre motivacional

"Con Django no solo aprendes a programar, aprendes a construir aplicaciones web profesionales que pueden crecer tanto como tus ideas 🚀."

👉 Próxima clase: Templates y contenido dinámico.

