



Es un lenguaje de programación *interpretado*, de tipado dinámico y multiparadigma (orientación a objetos, programación imperativa y funcional).

Creado por Guido van Rossum a finales del año 1991 y actualmente es administrado por la Python Software Foundation.

Posee un contador de referencias para poder liberar recursos cuando ya no son referenciados (usados).

Es un lenguaje altamente recomendado para dar los primeros pasos en el mundo de la programación.

Django



Framework de Python
para Aplicaciones Web

¿Qué es Django?

- Es un framework de Python para crear aplicaciones web que comenzó a desarrollarse en el año 2002 pero se publicó de forma oficial en el año 2005.
- Fomenta el desarrollo rápido y diseño limpio y pragmático (muy ligado a la resolución de problemas en la práctica).
- Muy apegado a la filosofía DRY (Don't Repeat Yourself).
- Basado en el patrón de arquitectura de software MVC.

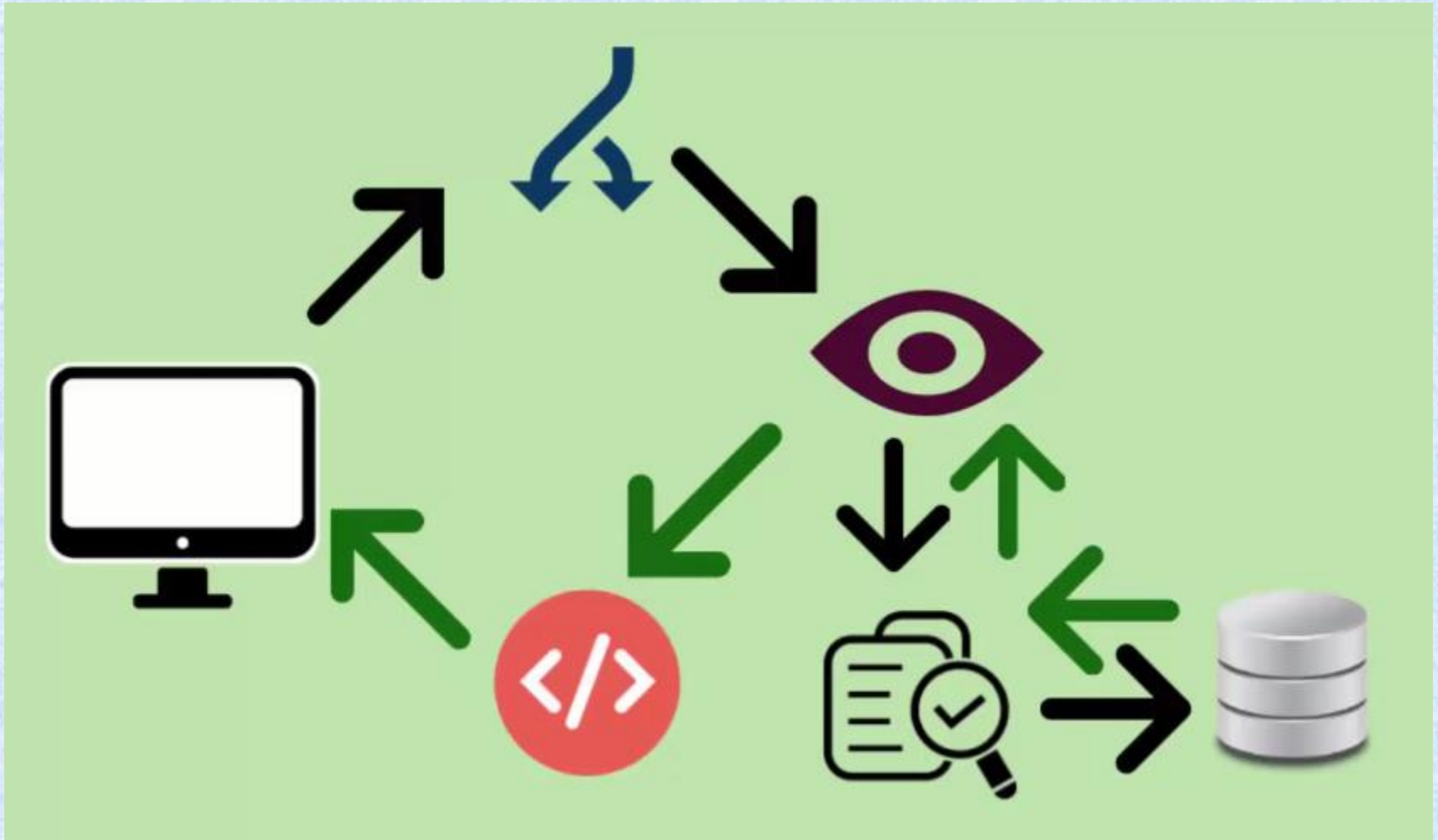
- Realmente, el patrón de arquitectura que sigue es el MTV (Modelo – Template - Vista).

Model - Template - View

- **Modelo:** Manipula los datos de la aplicación.
- **Template:** Decide cómo se van a mostrar los datos en el navegador.
- **View:** Decide cuáles datos va a mostrar el template.



¿Cómo funciona el patrón MTV (Model – Template – View)?



Estructura de un Proyecto en Django

Aplicación Web: Es el proyecto en sí.

Aplicación en Django: Es un módulo o funcionalidad del proyecto.

Las aplicaciones son las que contienen a los modelos.

Modelo (Model)

Es la fuente única y definitiva de los datos de nuestra aplicación que nos permite realizar operaciones con ellos.

Dicho de otra forma, un modelo se entiende como una **tabla** de la base de datos que soporta nuestra aplicación.

El gran beneficio que tenemos es el ORM (Object-Relational mapping), es decir, el mapeo objeto – relacional, del cual ya no tendremos que ocuparnos.

Todos nuestros modelos extienden (heredan) de la clase *django.db.models.Model*

Los modelos tienen **atributos** que se convierten en **campos** en las tablas de la base de datos.

Vista (View)

Es el elemento de nuestra aplicación donde ponemos la lógica de negocio así como las funciones necesarias de apoyo.

Aquí se procesan las peticiones o solicitudes (request) que accederán al Model (Model) para poder obtener o entregar datos.

La vista genera una respuesta, para el caso de Django se utiliza el método *render* para establecer la plantilla (template) de destino hacia dónde irán los datos.

Dicho de otra forma, aquí se establece la lógica necesaria para devolver una respuesta (y que datos conforma dicha respuesta) hacia el cliente que la solicita.

Plantilla (Template)

Es el elemento (documento HTML) que decide CÓMO se van a mostrar los datos devueltos por la vista.

Cada plantilla obtiene una respuesta proveniente de la vista correspondiente y contiene la estructura necesaria para presentar los datos en la forma necesaria utilizando estilos CSS o brindando dinamismo a través de JavaScript.

¡Gracias por su atención!

Oscar Alejandro Flavio García Fuentes

Sitio web: www.glcomputadoras.com

Email: UskoKruM2010@gmail.com / o.Oscar.-@hotmail.com

Teléfono: 996645436

www.youtube.com/UskoKruM2010