



Entrenamiento de Python-Django - 1

Base Django

Resumen: ¡Después de Python, Django!

Versión: 1.1

Contenido

I	Preámbulo	2
II	Reglas generales	3
III	Reglas específicas de hoy	4
IV	Ejercicio 00	5
V	Ejercicio 01	6
VI	Ejercicio 02	8
VII	Ejercicio 03	10
VIII	Presentación y evaluación por pares	12

Capítulo I

Preámbulo

Aquí hay una lista de monstruos que puedes encontrar vagando por una mazmorra en la tierra de Fangh:

- cualquier tipo de no-muerto.
- arañas gigantes.
- orcos y goblins.
- trolls en el subsuelo.
- hechiceros.
- guerreros malditos.
- ratas gigantes.
- una botella de aceite.
- papel higiénico.
- dos esponjas
- raviolos.

Capítulo II

Reglas generales

- Su proyecto debe realizarse en una máquina virtual.
- Su máquina virtual debe tener todo el software necesario para completar su proyecto. Estos software deben configurarse e instalarse.
- Puede elegir el sistema operativo que utilizará para su máquina virtual.
- Debe poder utilizar su máquina virtual desde una computadora del clúster.
- Debe utilizar una carpeta compartida entre su máquina virtual y su máquina host.
- Durante tus evaluaciones utilizarás esta carpeta para compartir con tu repositorio.

Sus funciones no deben cerrarse inesperadamente (fallo de segmentación, error de bus, doble liberación, etc.), salvo por comportamientos indefinidos. Si esto ocurre, su proyecto se considerará no funcional y recibirá un 0 en la evaluación.

Te animamos a crear programas de prueba para tu proyecto, aunque este trabajo no tenga que entregarse ni se califique. Esto te permitirá evaluar fácilmente tu trabajo y el de tus compañeros. Estas pruebas te resultarán especialmente útiles durante la defensa. De hecho, durante la defensa, puedes usar tus propias pruebas o las del compañero que estás evaluando.

Envía tu trabajo al repositorio Git asignado. Solo se calificará el trabajo en el repositorio Git. Si Deepthought se encarga de calificar tu trabajo, lo hará después de la evaluación por pares. Si se produce un error en alguna sección de tu trabajo durante la calificación de Deepthought, la evaluación se detendrá.

Capítulo III


Reglas específicas de hoy

- Todas las rutas relativas a una aplicación deben definirse en un archivo `urls.py` ubicado en una carpeta de esta aplicación.
- Cualquier formulario (clase derivada de `django.forms.Form`) debe estar ubicado en `forms.py` archivo de la aplicación con el que está relacionado.
- Cada página mostrada debe tener el formato correcto (se debe incluir un doctype, un par de etiquetas `html`, cuerpo y encabezado), gestión adecuada de caracteres especiales y sin visualizaciones extrañas.
- Hoy utilizarás el servidor de desarrollo predeterminado de Django proporcionado con `manage.py` utilidad.
- Solo las URL solicitadas específicamente deben devolver una página sin errores. Por lo tanto, Si sólo se solicita `/ex00`, `/ex00foo` debe devolver un error 404.

Las URL solicitadas deben funcionar con o sin barra final. Por lo tanto, si se solicita `/ex00`, `/ex00` y `/ex00/` deben funcionar ambos.

Capítulo IV

Ejercicio 00

	Ejercicio 00
Ejercicio 00: Primera página estática.	
Directorio de entrega: ex00/	
Archivos a entregar: requirements.txt, archivos y carpetas de su proyecto y aplicación Funciones permitidas:	
Todas las	
funcionalidades de Django.	

Con este ejercicio crearás tu primera página estática con Django.

Cree un virtualenv con python3, instale Django, cree un archivo requirements.txt que contenga las dependencias del proyecto (lo que está instalado en el virtualenv con pip) en la raíz del repositorio.

Iniciar un proyecto d05.


Iniciar una aplicación ex00.

Cree una página que permita búsquedas en la URL /ex00 de su sitio web. En esta página, recopile toda la información sobre la sintaxis de Markdown y titúlela "Ex00: Hoja de referencia de Markdown".

La plantilla utilizada se llamará index.html.

Capítulo V

Ejercicio 01

	Ejercicio 01
Ejercicio 01: Unas cuantas páginas más.	
Directorio de entrega: ex01/	
Archivos a entregar: requirements.txt, archivos y carpetas de su proyecto y aplicación.	
Funciones permitidas: todas las funcionalidades de Django.	

Crea las siguientes páginas en una segunda aplicación ex01:

- Título: "Ex01: Django, framework web."

URL: /ex01/django.

Descripción: En esta página, presenta brevemente Django y su historia.

- Título: "Ex01: Proceso de visualización de una página estática."

URL: /ex01/display

Descripción: En esta página se describe el proceso que provoca la visualización de una página web estática desde una plantilla simple pasando por una vista, desde la solicitud hasta la respuesta.

- Título: "Ex01: Motor de plantillas".

URL: /ex01/plantillas

Descripción: En esta página, describe el funcionamiento del motor de plantillas predeterminado de Django, así como el funcionamiento de:

- Bloques.
- Bucles para ... pulg.
- si hay estructuras de control.
- La visualización de las variables de contexto de paso.

El principio de no repetirse forma parte de la filosofía de Django. Para respetarlo, cree las páginas necesarias usando una plantilla base llamada `base.html`.

La plantilla `base.html` debe incluir:

- Un bloque de contenido en el cuerpo.
- Un bloque de estilo en la cabeza.
- Un bloque de título en la cabeza.

También cree una plantilla `nav.html` que contenga una barra de navegación que enumere los enlaces a cada una de las páginas del ejercicio.

Todas las páginas de este ejercicio deben basarse en la plantilla `base.html`. La plantilla `nav.html` debe estar incluida en cada una de las páginas.

Crea también dos archivos: `style1.css` y `style2.css`. El primero definirá el color del texto como azul, el otro como rojo. Deberás usar `style1.css` en todas las páginas, excepto en la página "Motor de plantillas", donde usarás `style2.css`.




Debes usar cada hoja de estilo solo una vez en todas tus plantillas para este ejercicio.



Necesitará utilizar `collectstatic` durante la evaluación.

Capítulo VI

Ejercicio 02

	Ejercicio 02
Ejercicio 02: Primera forma.	
Directorio de entrega: ex02/	
Archivos a entregar: requirements.txt, archivos y carpetas de su proyecto y aplicación.	
Funciones permitidas: todas las funcionalidades de Django.	

Crea una página accesible en la URL /ex02 en una nueva aplicación ex02.

Esta página contendrá 2 partes:

- Un formulario con un campo de texto y un botón Enviar.
- Una parte que contiene el historial de entrada.

Aquí está la regla respecto a la forma:

No debe codificar el campo del formulario. Para crear un formulario, usará la clase `django.forms.Form` de Django. La pasará como contexto a la función de renderizado de la plantilla.

Aquí están las reglas para la historia:

- El historial de entrada comenzará vacío.
- Para cada envío de texto en el formulario, deberás:
 - Incluya la entrada y su marca de tiempo al final de un archivo de registros. Si el archivo no existe, debe crearse automáticamente. Este archivo debe crearse en la carpeta o en la aplicación ex02.
 - Agrega la entrada al historial de la página con la fecha y hora del envío.


La ruta al archivo de registros (incluido el nombre del archivo) debe definirse en el archivo settings.py del proyecto. Por lo tanto, cada entrada enviada se incluirá en el historial de la página y en el archivo de registros, precedida por su marca de tiempo.

Los datos también deben ser persistentes. Si el servidor de desarrollo debe reiniciarse, no deben perderse.

Como regla general, si vuelves a mostrar la página, los datos se vuelven a mostrar siempre que estén guardados.

Capítulo VII

Ejercicio 03

	Ejercicio 03
Ejercicio 03: Cincuenta sombras de bic.	
Directorio de entrega: ex03/	
Archivos a entregar: requirements.txt, archivos y carpetas de su proyecto y aplicación.	
Funciones permitidas: todas las funcionalidades de Django.	

Crear una aplicación final ex03.

Muestra una página que contiene una tabla de 4 columnas x 51 líneas (una línea estará dedicada al nombre de la columna).

Accederá a esta página en esta URL /ex03.

Cada columna de la tabla tendrá un color diferente: negro, rojo, azul y verde.

Las cajas de mesa deberán tener las siguientes especificaciones:

Altura: 40 píxeles.

Ancho: 80 píxeles.

Color de fondo: un tono de color que coincide con la columna.

La tabla debe mostrar el tono de cada color para obtener un sombreado de 50 líneas.

Debes tener en cuenta las siguientes instrucciones:

- Su plantilla NO debe incluir los colores en letra dura. Los diferentes tonos deben ser generados en una vista y todos deben ser diferentes.
- Se autorizan un total de 4 parejas de etiquetas `<td></td>` en sus plantillas.
- Se autorizan un total de 4 parejas de etiquetas `<th></th>` en sus plantillas.
- Una tabla debe tener el siguiente formato:
 - Un par de etiquetas `<th></th>` por cuadro para los nombres de las columnas.
 - Un par de etiquetas `<tr></tr>` por línea de tono de color.
 - Una pareja de etiquetas `<td></td>` por caja de tonos de color.

Capítulo VIII

Presentación y evaluación por pares

Entrega tu tarea en tu repositorio de Git como de costumbre. Solo se evaluará el trabajo dentro de tus repositorios durante la defensa. No dudes en verificar los nombres de tus carpetas y archivos para asegurarte de que sean correctos.



El proceso de evaluación se realizará en el computador del evaluado.
grupo.