Un dibujo de un perro

Descripción generada automáticamente con confianza mediaLogotipo

Descripción generada automáticamente

Icono

Descripción generada automáticamente

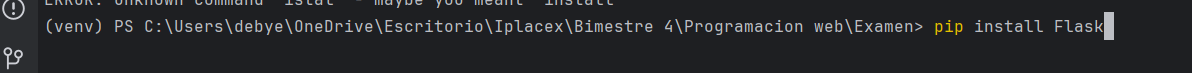
Evaluación 3

**2024-2B-PROGRAMACIÓN WEB/201-CED-QM**

**Debye Igor Verdejo**

**Instalando FLASK**

Abrimos la terminal en pycharm y ejecutamos el comando **pip install flask** esperamos que se complete el proceso



Para comprobar la instalación y versión de flask utilizamos el comando **python -m flask –version**

**Texto

Descripción generada automáticamente**

Crear carpeta del proyecto

Seleccionamos en crear nuevo proyecto, le asignamos nombre y elegimos la versión del intérprete de Python y nos aseguramos que tengamos habilitada la opción **Virtualenv** para el ambiente de trabajo   
Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Creamos las carpetas del proyecto, haciendo click secundario en la carpeta principal new-directory.

Creamos 2 carptas static y templates.

**Interfaz de usuario gráfica, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente**

En el archivo app.py (main) importamos flask y las funciones que usaremos (request y render\_template) **from flask import Flask, request, render\_template**

Iniciamos Flask y configuramos la ruta para la página index. (‘/’)

Texto

Descripción generada automáticamente

En la carpeta templates creamos un nuevo archivo HTML y lo nombramos index, ejercicio 1 y ejercicio 2

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

Configuramos el archivo index.html según el requerimiento

Para que se refleje correctamente el estilo css incorporado, en el head del documento hay que incorporar la etiqueta **<link rel="stylesheet" href="/static/css/styless.css">**  que hace referencia a la ubicación de nuestra hoja de estilos.

Texto

Descripción generada automáticamente

Hoja de estilos css, aquí agregamos las atributos para darle diseño a nuestro contenedor y al cuerpo de la página, les cambio el color de fondo para que se note diferencia, entre otros ajustes.

Los archivos CSS se guardan en la carpeta /static/css

Texto

Descripción generada automáticamente

Ejecutamos el archivo app.py haciendo click secundario y presionando run, o en la parte superior derecha en el icono verde de play.

Nos abre nuestro servidor local, clickeamos la ip y abrimos el proyecto en la pagina de incio

Texto

Descripción generada automáticamente

Nos abre en el navegador nuestro proyecto con los 2 botones

Interfaz de usuario gráfica, Diagrama

Descripción generada automáticamente

**Ejercicio 1**

En templeates creamos la pagina HTML ejercicio1 y la configuramos según el requerimiento, un formulario con campos para ingresar datos del cliente y de la factura, para calcular si se aplica algún descuento y el precio final.

Creamos un contenedor para todos los elementos.

Insertamos un titulo y bajo un formulario con la etiqueta **<form method="post" class="form">** es importante especificar aquí el **method="post"**  para la recopilación correcta de los datos( si no esta, no va a funcionar).

Creamos los Label e input asociados entre si y configuramos su tipo y lo hacemos que sea un campo obligatorio.

Texto

Descripción generada automáticamente

Bajo el formulario agregamos un contenedor div, que mostrara solo su contenido si el nombre y el total sin descuento tienen valores asosciados.

Abrimos el archivo app.py y configuramos la nueva página y sus funciones

**@app.route('/ejercicio1', methods=['GET', 'POST'])** Seleccionamos la ruta y agregamos los métodos para la obtención de los datos.

Recopilamos los datos de las variables del formulario con la función **request**  y realizamos la operación para el calculo de las variables solicitadas.

Texto

Descripción generada automáticamente

Ejecutamos, accedemos a la página 1 y realizamos pruebas con Nombre Felipe, 32 años y 10 Tarros

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

**Ejercicio 2**

Creamos una nueva pagina HTML en la carpeta templates y le asignamos el nombre ejercicio2

Creamos el formulario para el inicio de sesión, es este caso usamos el tipo “Password” para el input de la contraseña

Texto

Descripción generada automáticamente

En el archivo app.py agregamos la ruta y creamos la función para resolver el requerimiento.

En la ruta agregamos los métodos y recopilamos la información del formulario.

Creamos un diccionario para agregar clave : valor con el nombre de cada usuario

Y después realizamos las operaciones para validarlas en el programa entregando el mensaje correspondiente a cada solicitud.

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteEjecutamos y realizamos las pruebas

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Cargando archivos a GitHub (Carga de este Word en respositorio)**

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

**Autoevaluación**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Criterios de revisión del informe con la información recopilada:** | **Completamente logrado**  **(4)** | **Medianamente logrado**  **(3)** | **Parcialmente logrado**  **(2)** | **No logrado**  **(0)** |
| 1. Incorporé los elementos básicos de Python | x |  |  |  |
| 1. Codifiqué correctamente los requerimientos solicitados para el encargo. | x |  |  |  |
| 1. Utilicé correctamente los comandos de GIT para subir el proyecto a mi repositorio. | x |  |  |  |
| 1. Finalicé la actividad, comprendiendo cada uno de los ejercicios desarrollados | x |  |  |  |
| 1. Logré el objetivo planteado para esta actividad | x |  |  |  |
| **Puntaje Total** | **20 puntos** | | | |
| **Puntaje Obtenido** | **20 Puntos obtenidos** | | | |
|  |  | | | |