**CLASE GENERADORA DE CALENDARIOS HTML**

Implementar una clase que permita visualizar en el navegador el calendario ofrecido por la clase HTMLCalendar que ofrece el módulo *calendar* que viene por defecto en instalado en Python. Esta clase debe implementar dos funciones que abrirán el navegador automáticamente y mostrarán los calendarios pedidos. La primera función debe permitir observar el mes actual, o el mes que se le especifique del año actual. La segunda función debe permitir observar el calendario completo de un año ingresado, si no se especifica el año debe mostrar el calendario del año actual.

**Solución**

La forma más básica de visualizar contenido en el navegador es por medio de la implementación integrada de archivos HTML, CSS y Javascript. Donde los archivos HTML se encargan de definir la estructura visual del sitio web, los archivos CSS permiten darle estilo a esta estructura y los archivos en Javascript permite definir la funcionalidad de los elementos dentro del HTML, estos elementos pueden ser: botones, tablas, campos de texto, etc. Cuando se trabaja con la clase HTMLCalendar, lo que se encarga esta es generar un respectivo código en HTML que dibuja un calendario en el navegador donde se ejecute. El problema con esta clase es que si se desea usar el código generado por esta se debe copiar y pegar manualmente; por lo que para el desarrollo del ejercicio se implementa la generación de un archivo HTML y luego su posterior ejecución por medio del paquete webbrowser ofrecido por Python.

El método constructor de la clase es:



|  |  |
| --- | --- |
| self.\_\_mesesCod | Diccionario encargado de realizar la conversión numérica de los meses. Esto porque cuando se desea el calendario de un mes en específico el usuario ingresa el nombre de este, pero el método usado para dar formato al HTML de la clase HTMLCalendar recibe es el número asociado al mes. |
| self.\_\_añoActual | Variable que captura el año actual por medio del paquete datetime de Python. |
| self.\_\_mesActual | Variable que captura el mes actual por medio del paquete datetime de Python. |
| self.\_\_contenido | Variable donde se guarda el contenido HTML que define un calendario. |

Ahora, los métodos implementados para solucionar lo que se piden son dos, uno llamado verMes y otro verAño. En cada uno se puede especificar qué se desea ver. En el método verMes se decide qué mes del año actual se desea ver, y en el método verAño se ingresa el año deseado, si este se deja vacío generará el calendario del año actual. En ambos métodos se llama un método privado que lo hace es generar el archivo HTML que luego será ejecutado por medo de la función open del paquete webbrowser.



La lógica detrás de los diferentes métodos es simple, lo único a destacar son los métodos heredados de la clase madre HTMLCalendar. Estos métodos son formatyear y formatmonth, cada uno lo que hace es retornar un string que contiene todas las instrucciones HTML para generar, ya sea el calendario total del año o el calendario asociado al mes. Este string es almacenado entonces en el atributo contenido. La lógica empleada en ambos métodos establece que, si el usuario ingresa algún año o algún mes, este será el utilizado. Pero si no se ingresa nada, se tome el año y mes actual.

Luego se llama al método privado generarEjecutarHTML, este método genera y abre un archivo .html que lleva el nombre de index. Para esto último se emplea la estructura with junto a la función de Python open. Luego, en el archivo abierto se escribe las instrucciones HTML que se encuentran en el atributo contenido. Por último, por fuera de la estructura with se ejecuta el archivo html por medio de la función open.

Para finalizar, los paquetes importados en el código son:

