

Actividad 3

Herramienta alternativa: Pydoc

Se trata de un módulo de la librería estandar de Python que sirve igualmente para generar documentación.

Funciona de manera similar a Sphinx, es necesario añadir comentarios en las funciones y generar documentación mediante comandos. Este es un ejemplo de código:

```
import pandas as pd
import matplotlib.pyplot as plt
class finances:
    """Clase basada en la actividad 1 de Buenas Prácticas de Programación.

    """
    def read_df(path):
        """Lee el archivo .csv y lo pasa a DataFrame de Pandas.

        Args:
            path (str): path del archivo

        Raises:
            SystemExit: para la ejecución del script cuando ocurre un error de lectura.

        Returns:
            df(DataFrame): dataframe con los datos de salida.
        """
        try:
            df = pd.read_csv(path, sep = '\t') #'finanzas2020[1].csv'
            return df
        except Exception as error:
            print('No se ha podido leer el fichero ', error)
            raise SystemExit
```

En la línea de comandos dentro del directorio del documento:

```
>>> python -m pydoc -w FGPA3
```

Se genera la siguiente documentación en formato html:

Python: module FGPA3 x +

Archivo | C:/Users/FranciscoJoseGalvezP/Documents/Actividades/BuenasPP/a3/code/FGPA3.html

¿Quiere establecer Microsoft Edge como su explorador predeterminado? Establecer como predeterminado

Modules

[pandas](#) [matplotlib.pyplot](#)

Classes

[builtins.object](#)

[finances](#)

```
class finances(builtins.object)
    Clase basada en la actividad 1 de Buenas Prácticas de Programación.

    Methods defined here:

    calculos(df)
        Devuelve otro dataframe con los cálculos pedidos en el ejercicio.

        Args:
            df (dataframe): dataframe con los datos de entrada.

        Returns:
            df (DataFrame): dataframe con los datos de salida.

    check_empty(df)
        Comprueba que las columnas de los meses del año son correctas y si las columnas están vacías.

        Args:
            df(DataFrame): dataframe con los datos de entrada.

        Raises:
            SystemExit: para la ejecución del script cuando ocurre un error de columna.

        Returns:
            df(DataFrame): dataframe con los datos de salida.

    df_processing(df)
        Procesado del dataframe: corrige errores y sustituye valor por cero si es incomprensible.

        Args:
            df (dataframe): dataframe con los datos de entrada.

        Returns:
            df(DataFrame): dataframe con los datos de salida.

    read_df(path)
        Lee el archivo .csv y lo pasa a DataFrame de Pandas.

        Args:
            path (str): path del archivo

        Raises:
            SystemExit: para la ejecución del script cuando ocurre un error de lectura.

        Returns:
```

Sphinx

El código fuente queda adjunto a la entrega y el archivo html generado está en /build/index.html .

Documentación Finanzas

Navigation

Quick search

Go

Welcome to Finanzas's documentation!

Introducción

Documentación generada para la clase finanzas.

Finanzas

class FGPA3. **finanzas**

Clase basada en la actividad 1 de Buenas Prácticas de Programación.

calculos()

Devuelve otro dataframe con los cálculos pedidos en el ejercicio.

Args:

df (dataframe): dataframe con los datos de entrada.

Returns:

df (DataFrame): dataframe con los datos de salida.

check_empty()

Comprueba que las columnas de los meses del año son correctas y si las columnas están vacías.

Args:

df(DataFrame): dataframe con los datos de entrada.

Raises:

SystemExit: para la ejecución del script cuando ocurre un error de columna.

Returns:

df(DataFrame): dataframe con los datos de salida.

df_processing()

Procesado del dataframe: corrige errores y sustituye valor por cero si es incomprensible.

Args:

df (dataframe): dataframe con los datos de entrada.