Ejercicios Clase
-- Titanic–

1. Ordena el DataFrame del Titanic por la columna 'Age' de forma ascendente.

2. Ordena el DataFrame del Titanic por las columnas 'Pclass' y 'Age' de forma ascendente.

3. Crea una nueva columna llamada 'Adult' que sea True si la edad es mayor o igual a 18, y False de lo contrario.

4. Crea una nueva columna llamada 'Family_size' que sea la suma de las columnas 'SibSp' y 'Parch'.

5. Filtra el DataFrame para mostrar solo las filas donde 'Fare' esté entre 50 y 100, inclusive.

6. Crea una nueva columna llamada 'Class_high' que sea True si 'Pclass' es 1, y False de lo contrario.

7. Filtra el DataFrame para mostrar solo las filas donde 'Embarked' sea 'C' o 'Q'. (Usando ISIN)

8. Rellena los valores faltantes en la columna 'Age' con la media de esa columna.

9. Crea una nueva columna llamada 'Has_Cabin' que sea True si 'Cabin' no es nulo, y False de lo contrario.

10. Ordena el DataFrame del Titanic por 'Fare' de forma descendente.

11. Crea una nueva columna llamada 'Is_Alone' que sea True si 'Family_size' es 0, y False de lo contrario.

12. Filtra el DataFrame para mostrar solo las filas donde 'Age' esté entre 20 y 30, inclusive, y 'Pclass' sea 1 o 2. (Usando between e ISIN)

Ejercicios Clase
-- Titanic–

13. Crea una nueva columna llamada 'Fare_Category' que sea 'High' si 'Fare' es mayor que 50, 'Medium' si está entre 20 y 50, y 'Low' de lo contrario.

14. Rellena los valores faltantes en la columna 'Embarked' con el valor más frecuente.

15. Crea una nueva columna llamada 'Age_Group' que sea 'Child' si 'Age' es menor que 18, 'Adult' si está entre 18 y 60, y 'Senior' de lo contrario.

16. Ordena el DataFrame del Titanic por las columnas 'Pclass' de forma descendente y 'Fare' de forma ascendente.

17. Crea una nueva columna llamada 'Is_Female' que sea True si 'Sex' es 'female', y False de lo contrario.

18. Filtra el DataFrame para mostrar solo las filas donde 'Embarked' no sea 'S’.

19. Rellena los valores faltantes en la columna 'Cabin' con 'Unknown’.

20. Crea una nueva columna llamada 'Has_SibSp' que sea True si 'SibSp' es mayor que 0, y False de lo contrario.

21. Filtra el DataFrame para mostrar solo las filas donde 'Age' no esté entre 25 y 35, inclusive.

22. Crea una nueva columna llamada 'Fare_Normalized' que sea el resultado de restar la media de 'Fare' a 'Fare' y dividir entre 'Fare’.

23. Rellena los valores faltantes en la columna 'Embarked' con el valor 'C' si 'Pclass' es 1, 'Q' si es 2, y 'S' si es 3.

Ejercicios Clase
-- Ejercicio Grupal –

Contexto:
Tienes el dataset del Titanic cargado como data. Se requiere realizar un análisis y limpieza de datos de manera exhaustiva utilizando pandas para preparar el DataFrame para un análisis más detallado.

1. **Comprobación de valores nulos:**
   - Realiza una comprobación completa de valores nulos en el DataFrame. Crea un DataFrame booleano que indique la presencia de valores nulos en el DataFrame.

   - Cuenta el total de valores nulos en cada columna y el total de valores nulos en el DataFrame completo.

2. **Relleno de valores nulos:**
   - Rellena los valores nulos en la columna Age con la media de la columna.
   - Rellena los valores nulos en la columna Fare con un valor constante, como 100.
   - Utiliza la moda para rellenar valores nulos en la columna Embarked.
   - Aplica un método de relleno hacia adelante (ffill) en la columna Cabin para sustituir los valores nulos.
   - Rellena hacia atrás (bfill) los valores nulos en la columna Cabin.

3. **Limpieza de datos con regex:**
   - Limpia todas las columnas de texto (strings) para eliminar espacios en blanco no deseados y cualquier carácter especial (acentos o diacríticos) utilizando expresiones regulares con regex.
   - Convierte todos los nombres de las columnas a minúsculas y elimina espacios al inicio y al final.

Ejercicios Clase
-- Ejercicio Grupal –

4. **Filtrado avanzado de datos:**
   - Extrae solo las filas donde el rango de edad esté entre 18 y 60 años, y donde el valor de la columna Fare esté por encima del percentil 50.

   - Se debe crear una nueva columna llamada "Categoria_Edad" con condiciones específicas dentro de una función:
     - Si la edad es menor a 30 años, debe contener el valor "Joven".
     - Si la edad está entre 30 y 45 años, debe contener el valor "Adulto".
     - Si la edad es mayor a 45 años, debe contener el valor "Mayor".

5. **Análisis numérico:**
   - Ordena el DataFrame por el valor de la columna Fare en orden descendente y elimina duplicados basándote en la combinación de las columnas PassengerId y Pclass.
   - Crea una nueva columna llamada "Fare_Rank" que indique el rango de cada pasajero basado en su tarifa (Fare) utilizando un método de ranking en orden descendente.

6. **Cálculo de puntuación para cada pasajero:**
   - Crea una función personalizada llamada calcular_puntuacion. Esta función debe tomar como entrada la información de un pasajero individual (las columnas Age, Fare, Pclass, y Survived) y devolver una puntuación calculada utilizando la siguiente lógica:
     - Si el pasajero sobrevivió (Survived = 1), suma 5 puntos a la puntuación.
     - Si la edad es mayor o igual a 50, suma 4 puntos.
     - Si la tarifa (Fare) es mayor a 200, suma 3 puntos.
     - Si la clase del pasajero es 1ª clase, suma 2 puntos.
     - Si la clase del pasajero es 3ª clase, resta 2 puntos.

   - Muestra que pasajero tiene mayor puntuación

Ejercicios Clase
-- Ejercicio Grupal –

7. Pregunta adicional compleja:

Crear una nueva columna llamada Indice_Sobrevivencia que se calcule con las siguientes reglas:

1. Puntaje base:
   - Comienza con un puntaje igual al doble de la tarifa (fare * 2).

2. Modificaciones al puntaje base (por cada fila):
   - Restar 10 puntos si la edad es mayor de 50.
   - Sumar 15 puntos si el pasajero pertenece a la clase 1.
   - Restar 20 puntos si pertenece a la clase 3.
   - Multiplicar el puntaje por 1.2 si el pasajero es hombre y sobrevivió.
   - Dividir el puntaje por 2 si tiene más de 60 años y está en clase 3

3. Finalmente, clasificar a los pasajeros en "Alta", "Media" o "Baja" probabilidad de sobrevivencia según el valor del Indice_Sobrevivencia:
   - "Alta" si el índice es mayor de 200.
   - "Media" si está entre 100 y 200.
   - "Baja" si es menor de 100.