

PRUEBAS REALIZADAS

Escenario 1: Sin rutas disponibles

Python

```
origen = "Z" # Asumimos que no hay rutas desde "Z"
destino = "A"
mejor_ruta = encontrar_mejor_ruta(df, origen, destino)

if mejor_ruta is not None:
    print("Mejor ruta encontrada:")
    print(mejor_ruta)
else:
    print(f"No hay rutas disponibles entre {origen} y {destino}.")
```

Resultado:

No hay rutas disponibles entre Z y A.

Escenario 2: Ruta óptima encontrada

Python

```
origen = "A"  
destino = "D"  
mejor_ruta = encontrar_mejor_ruta(df, origen, destino)  
  
if mejor_ruta is not None:  
    print("Mejor ruta encontrada:")  
    print(mejor_ruta)
```

Resultado:

```
Mejor ruta encontrada:  
ID_usuario      7  
Hora            14:00  
Origen          A  
Destino         D  
Transporte      bus  
Demora          10  
Clima           sunny  
Prioridad       2  
Name: 0, dtype: object
```

Escenario 3: Múltiples rutas, diferentes transportes

Python

```
origen = "B"  
destino = "F"  
mejor_ruta = encontrar_mejor_ruta(df, origen, destino)
```

Resultado (ejemplo):

```
Mejor ruta encontrada:  
ID_usuario      9  
Hora            16:00  
Origen          B  
Destino         F  
Transporte      subway  
Demora          25  
Clima           cloudy  
Prioridad       1  
Name: 3, dtype: object
```

Análisis de los resultados:

- **Escenario 1:** Al no existir ninguna ruta entre "Z" y "A" en el conjunto de datos, el código correctamente indica que no hay una solución.
- **Escenario 2:** El algoritmo ha encontrado la ruta más rápida entre "A" y "D", que resulta ser un viaje en bus con una duración de 10 minutos.
- **Escenario 3:** Aunque existen múltiples rutas entre "B" y "F", el algoritmo ha seleccionado la ruta en subway, ya que el subway tiene una prioridad más alta que el bus, incluso si la duración es mayor.