

# **README {Metro Journey Planner} – Bonus Component**

## **1. Bonus Features Added**

I implemented extra functionality beyond the main assignment requirements. These additions improve user experience and give more detailed journey information.

### **BONUS FEATURES:**

1. Display of 4 subsequent metro arrival times after the next metro.
2. Fare calculation based on total travel time.
3. Extended additional metro line, including the Red Line.

## **2. Details of Additional Logic**

- Subsequent Metros:

After computing the next train time, the program continues to add intervals (4 min in peak hours or 8 min otherwise) and prints the next 4 metro timings.

This is repeated both at the starting station and the interchange station.

- Fare Calculation:

Fare =  $0.5 \times$  total minutes of travel

This was not part of the main assignment but was implemented as an enhancement.

- Red Line Support:

While the data file contains Magenta and Blue lines, I ensured correct functioning for Red Line operations too.

### **3. Why These Features Are Bonus Additions**

These functionalities were not asked in the main assignment but add value

- Commuters can plan better by seeing more metro timings.
- Fare estimation simulates a realistic metro system.
- Line extension testing demonstrates deeper system handling.

### **4. Additional Assumptions**

- Fare rate is fixed at ₹0.5 per minute.
- All subsequent metros follow the same frequency pattern as the next metro.

### **5. How to Run Bonus Features**

These features are automatically included in the same main Python file:

`metro_simulator.py`

No extra steps are needed.