# Vehicle Tracking Report

## 1. Project Overview

This project focuses on building a **real-time vehicle detection and tracking system** using **YOLO for object detection** and **ByteTrack for multi-object tracking**.
The goal is to detect vehicles in video footage, track them across frames, and generate insights such as vehicle counts.

**Key components:**

- `data/` → input video datasets
- `models/` → pretrained YOLO weights and tracking configs
- `results/` → generated tracking outputs
- `notebooks/` → exploratory analysis and experiments
- `README.md` → project documentation

---

## 2. Objective

The objective of this project is to:

- Implement a detection + tracking pipeline.
- Work with real-world traffic data.
- Produce reproducible results for vehicle monitoring and analysis.

---

## 3. Steps Performed

1. **Project Setup**
    - Created a Python environment.
    - Installed required dependencies.
2. **Model Integration**
    - Integrated YOLO for vehicle detection.
    - Connected ByteTrack for multi-object tracking.
3. **Running Experiments**
    - Processed sample traffic videos.
    - Generated outputs with vehicle bounding boxes and unique IDs.
4. **Results Storage**
    - Saved processed videos in the `results/` folder.
    - Documented findings in the report.

---

# 4. Observations / Results

- Vehicles are detected accurately with YOLO.
- ByteTrack maintains consistent IDs across frames.
- The system successfully counts and tracks vehicles over time.
- Project structure ensures modularity and easy extension.

---

# 5. Conclusion

This project demonstrates how **YOLO + ByteTrack** can be combined to build an effective vehicle tracking system. The workflow ensures clarity, reproducibility, and can be extended to other domains such as **pedestrian tracking, traffic flow monitoring, or smart city analytics**.