LLD PRACTICE SELF-VALIDATION CHECKLIST

1. REQUIREMENT VALIDATION

[ ] All functional requirements are covered.

[ ] Non-functional requirements considered (scalability, concurrency, etc.).

[ ] Edge cases handled (e.g., no spot available, invalid ticket).

2. ACTORS AND USE CASES

[ ] Identified all actors (User, Admin, System, etc.).

[ ] Defined what each actor can do.

[ ] Use cases are clear and mapped.

3. KEY ENTITIES AND DESIGN

[ ] All key entities/classes identified.

[ ] Relationships are correctly modeled (is-a / has-a).

[ ] SRP (Single Responsibility Principle) is followed.

[ ] Responsibilities of each class are well defined.

4. CLASS DIAGRAM REVIEW

[ ] Attributes and methods are realistic.

[ ] Composition, aggregation, and inheritance used correctly.

[ ] UML diagram is readable and clean.

5. OOP AND SOLID PRINCIPLES

[ ] OOP principles (Encapsulation, Inheritance, Polymorphism) used effectively.

[ ] SOLID principles are followed.

6. SCENARIO DRY RUN

[ ] Simulated core flows (e.g., park vehicle, unpark, fee calc).

[ ] No missing links in the flow.

[ ] Object collaboration is clear.

7. EXTENSIBILITY & SCALABILITY

[ ] Can easily add new vehicle types?

[ ] Can handle concurrent users?

[ ] Easy to plug in new rules or features?

8. OPTIONAL CODE CHECK

[ ] Implemented key classes and flow.

[ ] ParkingLotService or Manager class handles coordination.

[ ] Methods and naming are clean.

9. COMMUNICATION AND SUMMARY

[ ] Can explain the design clearly and logically.

[ ] Discussed trade-offs, patterns, and assumptions.

SCORE YOURSELF (out of 10 for each)

- Requirement coverage: ______ / 10
- Entity & Class Design: ______ / 10
- Code quality (if done): ______ / 10
- Design clarity and explanation: ______ / 10
- Total Score: ______ / 40

Tip: Track improvements across sessions!