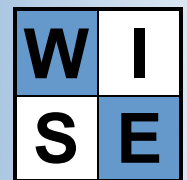
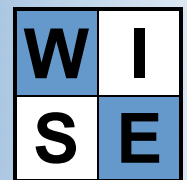


# DB2 UDB for z/OS: DB2 Locking & Concurrency for Developers



WISE LTD.

- COURSE CODE:** DZOLC1
- COURSE TITLE:** DB2 Locking & Concurrency for Developers
- ENVIRONMENT:** Non-data sharing
- AUDIENCE:** Application Developers (Designers and Programmers), Database Administrators, System Administrators and Capacity Planners.
- PREREQUISITES:** At least one (1) year of application development experience.
- DURATION:** 1 day
- SUMMARY:** Designed especially for Application Developers, this presentation's main focus is on how to design application processes that maximise concurrency while minimising locking overhead or CPU cycles in a non-data sharing environment.
- OBJECTIVES:** Upon completion of this presentation, the delegate should be able to make application-based locking choices that maximise concurrency and control deadlock/timeouts and 'hot spots'.



### 1. INTRODUCTION

- Locking Concepts
- Why locks are necessary
- Unit-of-work Concepts
- Two-phase COMMIT concepts
- Transaction locks
- Latches
- Claims/drains

### 2. LOCKING EFFECTS

- Suspensions
- Timeouts
- Deadlocks

### 3. APPLICATION DESIGN, GENERAL

- Data access patterns
- Commit frequency
- Retry logic
- When to open/close cursors
- BIND: ACQUIRE/RELEASE
- BIND: CURRENTDATA
- BIND: ISOLATION options
- Lock escalation
- Lock avoidance
- How to monitor locks

### 4. ONLINE (e.g., CICS)

- How to control 'hot spots'
- Row vs. Page locks
- Sequential number generation
- High volume inserts
- How to control 911s and 913s

### 5. BATCH DESIGN

- COMMIT frequency
- WITH HOLD
- FOR UPDATE

### 6. HOW TO MONITOR LOCKS

- EXPLAIN
- Traces (accounting reports)
- -DISPLAY DATABASE...

### 7. SOME USEFUL LOCKING DESIGN GUIDELINES