

Functional Programming

Variables are immutable

- Concurrent applications
 - All race conditions, deadlock conditions and concurrent update problems are due to mutable variables
- Immutability is not always practical
- Segregation of application into mutable and immutable components
 - Push as much as processing as possible into the immutable components
 - Event sourcing to store transactions, but not the state
 - Appropriate disciplines to protect mutable variables
 - Transactional memory, compare & swap algorithm

Design Principles