Review

IS 313
3.13.2003
Overview: Data

- Inside the application
  - Collections
- Outside the application
  - Database
- Getting/displaying
  - Swing
- Communicating over the network
  - Sockets
  - RMI
Overview: Architecture

- Inside the application
  - Collections
- Inside the enterprise
  - Database
- Interface to the user
  - Swing
- Systems integration
  - Sockets
  - RMI
Process

- 5 topics
- For each
  - a) what (I think) you should know
  - b) what (I think) you should be able to do
- If there are no questions
  - assume I’、“ right about (a) and (b)
  - next topic
Knowledge: Collections I

- Framework
  - Collection
  - List
  - Set
  - Map
  - Iterator
- Interface / Implementation distinction
- Implementations
  - ArrayList
  - HashSet, HashMap
  - TreeSet, TreeMap
Knowledge: Collections II

- Sorting
  - Comparable
  - Comparator
- Map collections
  - keySet()
  - values()
  - entrySet()
- Encapsulation
  - strategy for making task-specific collection objects
Knowledge: Skills I

- Write iteration loops through collections using an Iterator
  - for style
  - while style
- Write a Comparator class
- Implement the Comparable interface tomorrow
- Write a class that encapsulates a Collection
JDBC: Knowledge I

- Framework
  - Connection
  - Statement
  - ResultSet
  - PreparedStatement
- Factory pattern
- Prepared statements
  - Parameterized SQL
  - setXXX to set parameter values
JDBC: Knowledge II

- Updatable ResultSet
  - How to create
  - To modify data
    - `updateXXX`
    - `updateRow`
  - To insert data
    - `moveToInsertRow`
    - `updateXXX`
    - `insertRow`
Transactions

- The purpose of transactions
- AutoCommit
- commit
- rollback
JDBC: Skills

- Write code to execute an update query
  - doesn’t return result set
- Write code to execute an ordinary query
- Write code to iterate through a ResultSet
- Write code to modify database using a updatable ResultSet
- Write code to query database using a PreparedStatement
- Write code to update database using transactions
Swing: Knowledge I

- **Components**
  - JFrame
  - JDialog
  - JPanel
  - JScrollPane
  - JTable

- **Containment**
  - hierarchy of components
Swing: Knowledge II

- Layout Management
  - FlowLayout
  - BorderLayout
  -BoxLayout
  -GridLayout

- Model-view-controller pattern
  -TableModel
Swing: Knowledge III

- Event handling
  - internals: event handling thread
  - publish / subscribe pattern
  - Listener interfaces
    - ActionListener
    - MouseListener
  - Adapter classes
    - MouseAdapter
  - Anonymous class
    - external reference
Swing: Skills

- From interface sketch
  - identify components
  - create containment hierarchy
  - select layout management for each intermediate component

- Write an event handler using anonymous classes

- Given a data structure
  - write a TableModel to display it
Threads: Knowledge I

- Thread class
  - run method
  - Lifecycle
    - New
    - Runnable
    - Running
    - Non-runnable
    - Dead

- Runnable interface
Threads: Knowledge II

- Resource conflict
  - why this happens
  - synchronized methods
- Thread communication
  - why this is a problem
  - wait () / notify ()
- Stopping threads
  - interrupt
  - termination variable
  - close file or socket
Threads: Knowledge III

- Timers
  - java.util.Timer
    - Runnable object scheduled on timer thread
  - javax.swing.Timer
    - ActionListener handled on event handling thread
Threads: Skills

- Write code for thread termination
- Write a thread class and code that invokes it
- Write a Swing timer for a periodic UI task
Sockets: Knowledge I

- Client/server applications
- Protocol
  - system of interacting messages
- Socket
  - two-way data stream
- Java sockets
  - Socket
  - ServerSocket
Sockets: Knowledge II

- ServerSocket
  - `accept()`
  - `accept` returns Socket

- Socket
  - `create = connect`

- Multi-threaded server
  - Why
Socket: Skills

- Write run loop for multi-threaded server
- Write protocol handling code
RMI: Knowledge I

- Framework
  - Remote interface
  - Remote object
  - Server program
  - Client program
  - RMI Registry

- Serialization
  - why it is necessary
  - how to achieve
RMI: Knowledge II

- **Stub**
  - RMI Compilation

- **Activation**
  - alternative to server program
RMI: Skills

- Make a class Serializable
- Write a remote object
- Write a server program
- Write an RMI client
Final Exam

- Tuesday, 3/18, 2:45-5:00 pm