

Course Wrap-up



UNIVERSITY OF
TORONTO

The final

□ Comprehensive (covers the whole term), including:

- relational model
- RA
- SQL and JDBC
- XML and DTDs
- FD theory and normalization
- ER modelling and DB design
- JSON

The final

- You need to know the syntax of each language.
 - Exception: We don't care much about your Java syntax.
- You don't need to memorize function/method APIs.
 - We will provide what you need.
- Comments are not necessary, but may help us give you part marks.

Preparing for the final

- ☐ Re-solve parts of the assignments where you didn't get full marks or your partner lead.
- ☐ For topics you aren't fully confident in, re-do the lecture prep and in-class exercises.
- ☐ To hit on things you need to practise, make up
 - your own queries in RA & SQL
 - your own methods in JDBC.
- ☐ Solve old tests and finals.

The final

- You need 40% on the final to pass the course, but
 - If the exam is unexpectedly long or difficult, we will raise the marks on it
 - We apply that rule with great care

Theme: expressive power

- $RA \subseteq SQL \subseteq \text{Java}$
- DTDs (very limited!) \subseteq XML Schema
- DTDs vs DDL to express constraints
- You can sometimes do surprisingly much with what appears to be very little

Theme: expressive power

□ Expressive power vs computational complexity

- SQL general assertions vs intra-table constraints.
- SQL triggers are a tradeoff.

□ Expressive power vs *language* complexity

- RA vs SQL vs Java:
how complex are expressions/statements in the language; proofs about them?
- But is some of the difference just good/bad design?

Theme: tradeoffs

☐ materialized views vs virtual views

- save time: don't have to be recomputed on every use
- lose time: must be updated when base tables change
- lose space: must be stored

☐ indices

- save time: search and other operations are faster
- lose time: to build and maintain indices
- lose space: must be stored

☐ Rigidity of SQL vs flexibility of XML

☐ Normalized schema vs fast operations

☐ No redundancy vs preserve dependencies

CSC443

- “Database System Technology”
- Takes the perspective of the DBMS builder.
- Topics like:
 - indices; query optimization
 - managing storage; concurrency control
 - transaction management
 - tuning for performance
 - data mining, data warehousing

Trends in DB Research

- ☐ managing huge amounts of data: approximate querying, statistical methods, self-tuning, power management
- ☐ managing uncertainty
- ☐ data privacy and security
- ☐ different kinds of data, e.g., temporal, spatial, data from sensors, social network data
- ☐ new languages (e.g., declarative) and interfaces, visualization of data
- ☐ Check out the VLDB conference

Thanks!