Lesson 1: Key takeaways

Relational databases vs NoSQL databases

- Relational database <=> relational data model
  - Schema on write: data structures and its relationships are well defined and known
  - Primary Keys and Foreign keys
  - Sql language to join several tables
- NoSQL (Not Only SQL)
  - Schema on read: data structure takes sense on read, application must be aware of the schema while reading data
  - JSON documents commonly used, as well as Key-value pairs

Relational database simplified architecture

- Resources consumed by any software running on a computer: RAM, CPU, DISK, NETWORK
- Database vs Database instance
  - Database = datafiles + undo files + redo files + dictionary files + temporary files, **all on disk**
  - Database instance = background processes + memory structures. Many background processes are working together when running a relational database system.
- Data access mechanism: read, write
  - Data cache: memory is used mainly to cache most accessed data.
  - TEMP: temporary files used for temporary operations like sorting, hash joining, grouping.
  - Redo structures: any data written to the database datafiles is written first to REDO. REDO will be used to recover the database whenever a crash occurs.
  - Undo: before changing any data in the database, the “before” value is written to the UNDO structure. This enhances both read consistency and rollback capabilities.
- Read consistency explained, at conceptual level