**Work through the following materials this week:**

1. **Read Pramod Sadalage’s** [**NoSQL Databases: An Overview**](http://www.thoughtworks.com/insights/blog/nosql-databases-overview) **(circa 2012) and explain:**
   1. **the key motivations for considering non-relational database systems.**Allows developers to develop without having to convert in-memory structures to relational structures.  
      Relational databases were not designed to run efficiently on clusters.  
      Rise of web made more things use clusters.
   2. ***aggregate* data models.**A collection of data that we interact with as a unit. These units of data or aggregates form the boundaries for ACID operations with the database, Key-value, Document, and Column-family databases can all be seen as forms of aggregate-oriented database.  
      Makes distribution of data easier.
   3. ***polyglot* persistence.**Using multiple data storage technologies. Different data is best dealt with different storage.
   4. **the relevance of the *CAP theorem*.**CAP theorem which states that in any distributed system we can choose only two of consistency, availability or partition tolerance.   
      The CAP theorem states that if you get a network partition, you have to trade off availability of data versus consistency of data.   
      noSql gives you more tools to deal with the CAP limitations of databases, and which you want to prioritize.
   5. **the types of NoSQL databases.**Key-value, Document databases, column family stores, graph databases
   6. **when (and when not) to use NoSQL database systems.**To improve programmer productivity by using a database that better matches an application's needs.  
      To improve data access performance via some combination of handling larger data volumes, reducing latency, and improving throughput.
2. **An alternate source of this basic information is Martin Fowler’s** [**Introduction to NoSQL**](http://www.youtube.com/watch?v=qI_g07C_Q5I) **video.**
3. **Use “Getting Started with the Key/Value API”, Chapters 3–5, from the** [**Oracle NoSQL Database**](http://docs.oracle.com/cd/NOSQL/html/index.html) **documentation as a reference. Be sure to know how to:**
   1. **Compare and contrast *Major* and *minor* key components.**
   2. **Use the appropriate commands to write and retrieve records.**
4. **Be sure that you can compare and contrast relational data representations with key-value representations. Note that we will not use these higher-level features: the NoSQL Table API (We’ll focus on the lower-level key/value API); the Avro value schema mechanism (We’ll restrict ourselves to simple string values).**
5. **This practical article by Re Lai,** [**Enterprise Application Development Using Oracle NoSQL Database**](http://www.oracle.com/technetwork/articles/bigdata/oracle-nosqldb-appdev-1891870.html)**, is also helpful. It includes some material that we don't care about (i.e., NetBeans, UML, Avro value-handling) but the following sections are useful for the homework. Skim them now and be prepared to reference them later as needed.**
   1. **“Represent Key-Value Pairs” — Designing keys**
   2. **“CRUD” — Implementing the standard persistent data operations**
   3. **“Composite Keys” — Designing multi-element keys**
   4. **“Model Entities” — Modeling relational records in a KV store**
   5. **“Model Secondary Indexes” — Storing and retrieving sorted data**
   6. **“Model Multi-values and Relationships” — Modeling foreign key relationships**