|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Course Code:** CSE3009 | **Course Title: No SQL Data Bases** | **TPC** | **3** | | **2** | **4** |
| **Version No.** | **1.0** | | | | | |
| **Course Pre-requisites/ Co-requisites** | **CSE2007** | | | | | |
| **Anti-requisites (if any).** | **None** | | | | | |
| **Objectives:** | 1. This course will explore the origins of NoSQL databases and the characteristics that distinguish them from traditional relational database management systems. 2. This covers the architectures and common features of the main types of NoSQL databases (key-value stores, document databases, column-family stores, graph databases) 3. Finally, discuss the criteria that decision makers should consider when choosing between relational and non-relational databases and techniques for selecting the NoSQL database that best addresses specific use cases. | | | | | |
| **Expected Outcome:** | On completion of the course, students will have the ability to   1. Explain the detailed architecture, define objects, load data, query data and performance tune NoSQL databases 2. Define NoSQL, its characteristics, history and primary benefits using NoSQL Databases. 3. Define the major types of NoSQL databases including a primary use case and advantages/disadvantages of each type. 4. Analyze semi-structured data and choose an appropriate storage structure | | | | | |
| **Module No. 1** | **Introduction To NoSQL Concepts** | | | **6 Hours** | | |
| Data base revolutions: First generation, second generation, third generation, Managing Transactions and Data Integrity, ACID and BASE for reliable database transactions, Speeding performance by strategic use of RAM, SSD, and disk, Achieving horizontal scalability with database sharding, Brewer’s CAP theorem. | | | | | | |
| **Module No. 2** | **NoSQL Data Architecture Patterns** | | | **8 Hours** | | |
| NoSQL Data model: Aggregate Models- Document Data Model- Key-Value Data Model- Columnar Data Model, Graph Based Data Model – Graph Data Model, NoSQL system ways to handle big data problems, Moving Queries to data, not data to the query, hash rings to distribute the data on clusters, replication to scale reads, Database distributed queries to data nodes. | | | | | | |
| **Module No. 3** | **Key –Value Data Stores** | | | **8 Hours** | | |
| From array to key –value databases, Essential features of key – value Databases, Properties of keys, Characteristics of Values, Key-Value Database Data Modeling Terms, Key-Value Architecture and implementation Terms, Designing Structured Values, Limitations of Key- Value Databases, Design Patterns for Key-Value Databases, Case Study: Key-Value Databases for Mobile Application Configuration | | | | | | |
| **Module No. 4** | **Document Oriented Database** | | | **7 Hours** | | |
| Document, Collection, Naming, CRUD operation, querying, indexing, Replication, Sharding, Consistency Implementation: Distributed consistency, Eventual Consistency, Capped Collection, Case studies: document oriented database: MongoDB and/or Cassandra | | | | | | |
| **Module No. 5** | **Columnar Data Model** | | | **8 Hours** | | |
| Data warehousing schemas: Comparison of columnar and row-oriented storage, Column-store Architectures: C-Store and Vector-Wise, Column-store internals and, Inserts/updates/deletes, Indexing, Adaptive Indexing and Database Cracking. Advanced techniques: Vectorized Processing, Compression, Write penalty, Joins, Group-by, Aggregation and Arithmetic Operations, Case Studies | | | | | | |
| **Module No. 6** | **Data Modeling With Graph** | | | **8 Hours** | | |
| Comparison of Relational and Graph Modeling, Property Graph Model Graph Analytics: Link analysis algorithm- Web as a graph, PageRank- Markov chain, page rank computation, Topic specific page rank (Page Ranking Computation techniques: iterative processing, Random walk distribution Querying Graphs: Introduction to Cypher, case study: Building a Graph Database Application- community detection | | | | | | |
| **Text Books**   1. Guy Harrison, “Next Generation database: NoSQL and Big data”, A Press ,2015 2. Ted Hills , “NoSQL and SQL Data Modeling: Bringing Together Data, Semantics, and Software”, Technics Publications,2016 | | | | | | |
| **References**   1. Daniel Abadi, Peter Boncz, Stavros Harizopoulos, Stratos Idreaos, Samuel Madden, “The Design and Implementation of Modern Column-Oriented Database Systems”, Now Publishers,2013 | | | | | | |
| **Lab Exercises**   1. Import the Hubway data into Neo4j and configure Neo4j. Then, answer the following questions using the Cypher Query Language:   a) List top 10 stations with most outbound trips (Show station name and number of trips)  b) List top 10 stations with most inbound trips (Show station name and number of trips)  c) List top 5 routes with most trips (Show starting station name, ending station name and number of trips) (4) List the hour number (for example 13 means 1pm -2pm) and number of trips which start from the station "B.U. Central"  d) List the hour number (for example 13 means 1pm -2pm) and number of trips which end at the station "B.U. Central"   1. The flight data can be found at <http://stat-computing.org/dataexpo/2009/thedata.html>   You need to download just one year and from there you can sample a subset of at least 10000 records. You can use the data from a full year if you want but we recommend using a smaller dataset for simplicity. Hint: If you need to unzip the data file, you can use the command: bzip2 –d datafile from a terminal. For example, for the 2008, you download the file and unzip it using: bzip2 -d 1987.csv.bz2. The airport data can be found at  <http://stat-computing.org/dataexpo/2009/supplemental-data.html>  1) Download the flight dataset and airport dataset.  (2) Clean the dataset (for example: remove columns you do not need, remove records with missing information, remove duplicate records and so on).  (3) Give the header to csv files  (4) Import the data into Neo4j.  (5) Write the queries to answer following questions:  (5.1) List top 10 airports with most outbound flights.  (5.2) List top 10 airports with most inbound flights.  (5.3) List top 5 routes with most flights in weekdays.  (5.4) List top 5 routes with most flights in weekends.  (5.5) List the hour number (for example 13 means 1pm -2pm) and number of flights, which depart from a specific airport in your data (e.g., Boston Logan Airport).  (5.6) List the hour number (for example 13 means 1pm -2pm) and number of flights, which arrive at specific airport in your data (e.g., Boston Logan Airport).  In your report, you should answer the following questions:  (a) List the year of the flights that you downloaded and prepared for this assignment. You can get a sample set from one-year data. However, the number of flights cannot be smaller than 10k.  (b) Describe how you clean the data (Which columns you remove and why? Which rows you remove and why?). Hint: You can clean your data by writing a small program in Java, Python, C, Matlab or any kind of programming language.  (c) Describe the header you give to the csv files.  (d) Write down the command for importing data.  (e) Write and execute the queries from step (5) above.   1. Download a zip code dataset at   <http://media.mongodb.org/zips.json>  Use mongoimport to import the zip code dataset into MongoDB.  After importing the data, answer the following questions by using aggregation pipelines:  (1) Find all the states that have a city called "BOSTON".  (2) Find all the states and cities whose names include the string "BOST".  (3) Each city has several zip codes. Find the city in each state with the most number of zip codes and rank those cities along with the states using the city populations.  (4) MongoDB can query on spatial information.  Assume we have a spatial position as [-72, 42], and in the range of 2 (it can be [-71.5, 41.5] or [-72.5, 42.5] or somewhere else), there may exist a number of zip codes . Try to find the states in that range. You should return the total populations and the number of cities of each state in that range. Rank the states based on the number of cities.  (5) Consider a certain rectangular area, in which the vertices are [ -80 , 30 ] , [ -90 ,30 ] , [ -90 , 40 ] and [ -80 , 40 ]. Find and report the top 10 largest cities (by population) in this area.   1. Create a database that stores road cars. Cars have a manufacturer, a type. Each car has a maximum performance and a maximum torque value. Do the following:   Test Cassandra’s replication schema and consistency models.   1. Network Partition without Replication 2. Network Partition with Replication and Weak Consistency 3. Network Partition with Replication and Quorum Consistency 4. Cars have different powertrains. Each type can be described with different parameters:   Internal combustion engine: fuel type, displacement, maximum torque, maximum power.Electric motor: maximum torque, maximum power Both: all of the above and the combined maximum torque and power values Construct the class hierarchy for different powertrain types Extend the cars column family to store the powertrain of each car.  Write a query that collects the cars with an internal combustion engine or an electric motor.   1. Master Data Mangement using Neo4j   Manage your master data more effectively  The world of master data is changing. Data architects and application developers are swapping their relational databases with graph databases to store their master data. This switch enables them to use a data store optimized to discover new insights in existing data, provide a 360-degree view of master data and answer questions about data relationships in real time.   1. Optimization of Customer Experience with Real-time Recommendations using Neo4j 2. The operational intelligence case studies describe applications that collect machine generated data from logging systems, application output, and other systems using mongoDB. 3. The product data management case studies address aspects of applications required for building product catalogs, and managing inventory in e-commerce systems (use MongoDB) 4. the content management case studies introduce basic patterns and techniques for building content management systems using MongoDB. 5. ShoppingMall case study using cassendra, where we have many customers ordering items from the mall and we have suppliers who deliver them their ordered items. 6. Key-Value Databases for Mobile Application Configuration   **Projects**  Projects may be given as group projects  The following list are the sample projects that can be given to students to be implemented:   1. Analyzing and Visualizing social networks like Facebook and twitter using NoSQL Databases. 2. Using Sample datasets from <http://www.rdatamining.com/resources/data>,UCLA Repository, kaggle dataset etc., and analyzing those using NoSQL databases. 3. Twitter provides a fire hose of data. Automatically filtering, aggregating, analyzing such data can allow a way to harness the full value of the data, extracting valuable information. The idea of this project is investigating stream processing technology to operate on social streams. 4. Project on Combining Database management and Cloud storage system. 5. CarTel. In the CarTel project, we are building a system for collecting and managing data from automobiles. There are several possible CarTel related projects:   a) One of the features of CarTel is a GUI for browsing geo‐spatial data collected from cars. Primitive interface for retrieving parts of the data that are of interest, but developing a more sophisticated interface or query language for browsing and exploring this data would make a great project. It collects relatively sensitive personal information about users location and  driving habits. Protecting this information from casual browsers, insurance companies, or other undesired users is important. However, it is also important to be able to combine different users data together to do things like intelligent route planning or vehicle anomaly  detection. The goal of this project would be to find a way to securely perform certain types of aggregate queries over CarTel data without exposing personally identifiable information. | | | | | | |
| **Mode of Evaluation** | **Practice Tests-20%, Continuous Assessment Tests-60%, Practical Assesment-20%**  Practice Tests - Cumulative for 16 Weeks 20%  Continuous Assessment Test-1 20%  Continuous Assessment Test-2 20%  Continuous Assessment Test-3 20%  Practical Assessment (Mini Project) 20% | | | | | |
| **Recommended by the Board of Studies on** | 06.07.2018 | | | | | |
| **Date of Approval by the Academic Council** | 2nd Academic Council 21.07.2018 | | | | | |