**Date: 15/01/2024**

**Batch: S1,S2,S3,S4,S5,S6,S7,S8**

**Experiment No. 2: Study Commercial & Open-source Database**

**Walchand College of Engineering, Sangli**

**Department of Computer Science and Engineering**

**Course: Database Engineering Lab (**6CS274**) Year: 2023-24**

**Course Teacher: Ms. Sonali S. Rokade**

1. Enlist Database systems.
2. Enlist the versions of Oracle.
3. What is the difference between i,g & c from 9i, 10g and 12c respectively.
4. Study following databases .

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Roll No.** | **Database** |
| **1** | 22110053 | 1010data |
| **2** | 22510001 | SQLBase |
| **3** | 22510003 | Versant Object Database |
| **4** | 22510004 | Cubrid |
| **5** | 22510005 | PrestoDB |
| **6** | 22510006 | Geode |
| **7** | 22510007 | MySQL |
| **8** | 22510008 | Microsoft SQL Server |
| **9** | 22510009 | PostgreSQL |
| **10** | 22510010 | MongoDB |
| **11** | 22510012 | DB2 |
| **12** | 22510013 | Microsoft Access |
| **13** | 22510014 | Redis |
| **14** | 22510015 | Cassandra |
| **15** | 22510016 | Elasticsearch |
| **16** | 22510017 | SQLite |
| **17** | 22510018 | Teradata |
| **18** | 22510019 | ORACLE |
| **19** | 22510020 | SAP Adaptive Server |
| **20** | 22510021 | Splunk |
| **21** | 22510022 | HBase |
| **22** | 22510023 | MariaDB |
| **23** | 22510025 | FileMaker |
| **24** | 22510027 | Hive |
| **25** | 22510028 | SAP HANA |
| **26** | 22510029 | Neo4j |
| **27** | 22510030 | Amazon DynamoDB |
| **28** | 22510031 | Couchbase |
| **29** | 22510033 | Memcached |
| **30** | 22510034 | Informix |
| **31** | 22510035 | Microsoft Azure SQL Database |
| **32** | 22510037 | Vertica |
| **33** | 22510038 | CouchDB |
| **34** | 22510039 | Amazon CloudSearch |
| **35** | 22510040 | Firebird |
| **36** | 22510041 | Netezza |
| **37** | 22510042 | Impala |
| **38** | 22510043 | Microsoft Azure Cosmos DB |
| **39** | 22510044 | Amazon Redshift |
| **40** | 22510045 | Google BigQuery |
| **41** | 22510046 | Spark SQL |
| **42** | 22510047 | MarkLogic |
| **43** | 22510048 | Oracle Essbase |
| **44** | 22510049 | dBASE |
| **45** | 22510050 | Greenplum |
| **46** | 22510051 | InfluxDB |
| **47** | 22510052 | Hazelcast |
| **48** | 22510053 | Ehcache |
| **49** | 22510054 | Riak KV |
| **50** | 22510055 | OrientDB |
| **51** | 22510056 | Interbase |
| **52** | 22510057 | Sphinx |
| **53** | 22510058 | Titan |
| **54** | 22510059 | Firebase Realtime Database |
| **55** | 22510060 | Ingres |
| **56** | 22510061 | HyperSQL |
| **57** | 22510062 | H2 |
| **58** | 22510063 | Microsoft Azure SQL Data Warehouse |
| **59** | 22510064 | Realm |
| **60** | 22510065 | Aerospike |
| **61** | 22510067 | RethinkDB |
| **62** | 22510070 | SAP IQ |
| **63** | 22510072 | Derby |
| **64** | 22510076 | SAP SQL Anywhere |
| **65** | 22510077 | Cloudant |
| **66** | 22510078 | Microsoft Azure Search |
| **67** | 22510079 | Jackrabbit |
| **68** | 22510080 | Accumulo |
| **69** | 22510081 | Adabas |
| **70** | 22510082 | MemSQL |
| **71** | 22510086 | Kdb+ |
| **72** | 22510089 | RavenDB |
| **73** | 22510090 | Amazon Aurora |
| **74** | 22510092 | ArangoDB |
| **75** | 22510093 | Microsoft Azure Table Storage |
| **76** | 22510094 | Google Cloud Datastore |
| **77** | 22510095 | OpenEdge |
| **78** | 22510096 | RRDtool |
| **79** | 22510097 | Oracle NoSQL |
| **80** | 22510099 | Algolia |
| **81** | 22510101 | UniData,UniVerse |
| **82** | 22510102 | Caché |
| **83** | 22510103 | Apache Drill |
| **84** | 22510105 | Amazon CloudSearch |
| **85** | 22510106 | Oracle Rdb |
| **86** | 22510107 | MaxDB |
| **87** | 22510108 | PouchDB |
| **88** | 22510109 | Oracle Berkeley DB |
| **89** | 22510110 | Ignite |
| **90** | 22510111 | TimesTen |
| **91** | 22510112 | Oracle Coherence |
| **92** | 22510113 | Amazon CloudSearch |
| **93** | 22510114 | Google Search Appliance |
| **94** | 22510115 | Graphite |
| **95** | 22510116 | SAP Advantage Database Server |
| **96** | 22510117 | Amazon SimpleDB |
| **97** | 22510118 | Teradata Aster |
| **98** | 22510119 | Infinispan |
| **99** | 22510120 | MapD |
| **100** | 22510121 | Percona Server |
| **101** | 22510122 | LevelDB |
| **102** | 22510123 | Jena |
| **103** | 22510124 | CloudKit |
| **104** | 22510125 | IBM dashDB |
| **105** | 23520001 | Google Cloud Firestore |
| **106** | 23520002 | 4D |
| **107** | 23520003 | OpenTSDB |
| **108** | 23520004 | Datameer |
| **109** | 23520005 | VoltDB |
| **110** | 23520006 | Datomic |
| **111** | 23520007 | jBASE |
| **112** | 23520008 | Virtuoso |
| **113** | 23520009 | MonetDB |
| **114** | 23520010 | IMS |
| **115** | 23520011 | RocksDB |
| **116** | 23520012 | EnterpriseDB |
| **117** | 23520013 | PrestoDB |

**Course Coordinator**

**Ms. Sonali S. Rokade**

Database assigned : Realm

**1.Enlist Database Systems:**

**Ans.**1. MySQL:

- An open-source relational database management system (RDBMS).

2. PostgreSQL:

- An open-source object-relational database system known for its extensibility and standards compliance.

3. Microsoft SQL Server:

- A relational database management system developed by Microsoft.

4. Oracle Database:

- A proprietary multi-model database management system developed by Oracle Corporation.

5. SQLite:

- A self-contained, serverless, and zero-configuration database engine.

6. MongoDB:

- A NoSQL document-oriented database, commonly used for handling large amounts of unstructured data.

7. Cassandra:

- A distributed NoSQL database designed to handle large amounts of data across many commodity servers without a single point of failure.

8. CouchDB:

- A NoSQL database that uses a document-oriented model, featuring a distributed architecture.

9. Redis:

- An in-memory data structure store often used as a caching mechanism.

10. Neo4j:

- A graph database management system, optimized for storing and querying graph data.

11. MariaDB:

- An open-source relational database management system, created by the original developers of MySQL.

12. Amazon DynamoDB:

- A fully managed NoSQL database service provided by Amazon Web Services (AWS).

13. Google Cloud Firestore:

- A NoSQL document database provided as a part of the Google Cloud Platform.

14. IBM Db2:

- A family of data management products, including a database server, developed by IBM.

15. Apache Derby

- An open-source relational database implemented entirely in Java.

**2.Enlist the versions of ORACLE:**

**Ans.** 1. Oracle Database 7:

- Released in 1992.

2. Oracle Database 8:

- Introduced in 1997.

- Included features like the Oracle8i for the internet.

3. Oracle Database 9i

- Released in 2001.

- "i" stood for internet, indicating the database's support for web-based technologies.

4. Oracle Database 10g:

- Released in 2003.

- "g" stood for grid computing, emphasizing the ability to manage databases as a grid.

5. Oracle Database 11g:

- Released in 2007.

- Focused on grid computing, consolidation, and improved management capabilities

6. Oracle Database 12c:

- Released in 2013.

- "c" stood for cloud, indicating increased support for cloud computing and multi-tenancy.

7. Oracle Database 18c:

- Released in 2018.

- Introduced a new naming convention, moving away from version numbers to a yearly release model.

8. Oracle Database 19c:

- Released in 2019.

- Followed the annual release model, continuing the 18c innovations.

**3.What is the difference I,g & c from 9i,10g and 12c respectively.**

**Ans.** The letters "i," "g," and "c" in Oracle Database versions 9i, 10g, and 12c represent different focuses or features of each release:

1. Oracle Database 9i:

- The "i" in Oracle Database 9i stood for "Internet."

- Oracle 9i, released in 2001, emphasized its support for internet computing, providing features and enhancements for better web-based application development.

2. Oracle Database 10g:

- The "g" in Oracle Database 10g stood for "Grid."

- Oracle 10g, released in 2003, focused on grid computing. It introduced features to manage databases as a grid and emphasized better scalability, availability, and performance.

3. Oracle Database 12c:

- The "c" in Oracle Database 12c stands for "Cloud."

- Oracle 12c, released in 2013, shifted the focus towards cloud computing. It introduced features such as pluggable databases, designed to make it easier to consolidate databases and manage them in a cloud environment. It also aimed at providing better support for multi-tenancy.

**4.Study following databases**.

**Ans.-**

**Advantages of Realm Database:**

**1. Performance:**

- Realm is known for its high-performance characteristics, offering faster read and write operations compared to some other mobile databases.

**2. Real-time Data Sync:**

- Realm supports real-time data synchronization, allowing multiple devices to work with the same data and receive updates in real-time.

**3. Cross-Platform Compatibility:**

- Realm provides support for various platforms, including iOS, Android, Xamarin, React Native, and more. This makes it suitable for cross-platform development.

**4. Object-Oriented Model:**

- Realm uses an object-oriented data model, where data is represented as objects in code. This makes it more intuitive for developers who work with object-oriented programming.

**5. Offline Support:**

- Realm allows for offline data access, enabling applications to continue functioning even when the device is not connected to the internet.

**6. Ease of Use:**

- The simplicity of Realm's API and the ease of integration with mobile development frameworks make it a convenient choice for developers.

**Disadvantages of Realm Database:**

**1. Limited Query Capabilities:**

- Realm's query language is more straightforward compared to traditional relational databases. While it covers many common use cases, complex queries may be challenging to express.

**2. Learning Curve:**

- Developers accustomed to traditional relational database concepts may experience a learning curve when adapting to Realm's object-oriented model and query language.

**3. Limited Data Analysis and Reporting:**

- If your application requires extensive data analysis and reporting, Realm may not be the best choice, as it is optimized for real-time data access rather than complex analytics.

**4. Size Limitations:**

- Realm has limitations on the maximum size of its database files. This might be a consideration for applications dealing with very large datasets.

**5. Community and Ecosystem:**

- While Realm has a growing community, it may not have the extensive ecosystem and community support that some other database solutions enjoy.

**6. Pricing Model:**

- Depending on your usage and requirements, Realm's pricing model might be a consideration. Large-scale or enterprise applications may have associated costs.

­­­­­