Дисциплина «ПРОЕКТИРОВАНИЕ ВЫСОКОНАГРУЖЕННЫХ СИСТЕМ ХРАНЕНИЯ ДАННЫХ»

(лекций 24 часа; практических занятий 24 часа; форма контроля - комбинированный экзамен; курсовая работа)

Лекции:

Колесник Людмила Владимировна

Практические занятия:

Коваленко Андрей Иванович

Модель данных

Модель данных представляет из себя теорию, инструмент моделирования, и включает в себя, как минимум, три аспекта:

а) аспект структуры, который включает в себя методы описания типов и логических структур в базе данных;

б) аспект манипуляции, который включает в себя методы манипулирования данных;

в) аспект целостности включает в себя методы описания и поддержки целостности базы данных.

Основные модели данных

– иерархическая модель;
– сетевая модель;
– реляционная модель;
– документно-ориентированная;
…
…
…

Требования, которые предъявляются к СУБД

– атомарность;
– согласованность;
– изолированность;
– надежность.

Рейтинг СУБД от TIOBE Software (2019-2020 гг.)

Rank			DBMS	Database Model	Score		
Jan 2020	Dec 2019	Jan 2019			Jan 2020	Dec 2019	Jan 2019
1.	1.	1.	Oracle	Relational, Multi-model	1346.68	+0.29	+77.85
2.	2.	2.	MySQL	Relational, Multi-model	1274.65	-1.01	+120.39
3.	3.	3.	Microsoft SQL Server	Relational, Multi-model	1098.55	+2.35	+58.29
4.	4.	4.	PostgreSQL	Relational, Multi-model	507.19	+3.82	+41.08
5.	5.	5.	MongoDB	Document, Multi-model	426.97	+5.85	+39.78
6.	6.	6.	IBM Db2	Relational, Multi-model	168.70	-2.65	-11.15
7.	7.	↑ 8.	Elasticsearch	Search engine, Multi-model	151.44	+1.19	+8.00
8.	8.	↓ 7.	Redis	Key-value, Multi-model	148.75	+2.51	-0.27
9.	9.	9.	Microsoft Access	Relational	128.58	-0.89	-13.04
10.	↑ 11.	10.	SQLite	Relational	122.14	+1.78	-4.66
11.	↓ 10.	11.	Cassandra	Wide column	120.66	-0.04	-2.32
12.	12.	12.	Splunk	Search engine	88.67	-1.85	+7.25
13.	13.	13.	MariaDB	Relational, Multi-model	87.45	+0.66	+8.63
14.	14.	↑ 15.	Hive	Relational	84.24	-1.81	+14.33
15.	15.	↓ 14.	Teradata	Relational, Multi-model	78.29	-0.21	+2.10

Темы практических занятий (лабораторных работ)

ПРОЕКТИРОВАНИЕ ВЫСОКОНАГРУЖЕННЫХ БАЗ ДАННЫХ НА ПЛАТФОРМЕ СУБД MYSQL С ИСПОЛЬЗОВАНИЕМ ТАБЛИЦ ТИПА MYISAM И INNODB.

1 РАЗРАБОТКА SQL-ЗАПРОСОВ С УЧЕТОМ СПЕЦИФИКИ ВЫСОКОНАГРУЖЕННЫХ БАЗ ДАННЫХ НА ПЛАТФОРМЕ СУБД MYSQL.

2 РЕАЛИЗАЦИЯ ПРОГРАММНОЙ ЛОГИКИ И ОБЕСПЕЧЕНИЕ ЦЕЛОСТНОСТИ СВЯЗЕЙ С ПОМОЩЬЮ ТРИГГЕРОВ ДЛЯ ВЫСОКОНАГРУЖЕННЫХ БАЗ ДАННЫХ НА ПЛАТФОРМЕ СУБД MYSQL.

3 СОЗДАНИЕ И ИСПОЛЬЗОВАНИЕ ПРЕДСТАВЛЕНИЙ ДЛЯ ВЫСОКОНАГРУЖЕННЫХ БАЗ ДАННЫХ НА ПЛАТФОРМЕ СУБД MYSQL.

4 СОЗДАНИЕ ПРОЦЕДУР И ФУНКЦИЙ ДЛЯ ВЫСОКОНАГРУЖЕННЫХ БАЗ ДАННЫХ НА ПЛАТФОРМЕ СУБД MYSQL.

5 ОПТИМИЗАЦИЯ SQL-ЗАПРОСОВ ВЫСОКОНАГРУЖЕННЫХ БАЗ ДАННЫХ НА ПЛАТФОРМЕ СУБД MYSQL.

6 СОЗДАНИЕ ТРАНЗАКЦИЙ ДЛЯ ВЫСОКОНАГРУЖЕННЫХ БАЗ ДАННЫХ НА ПЛАТФОРМЕ СУБД MYSQL.

ПРОЕКТИРОВАНИЕ ВЫСОКОНАГРУЖЕННЫХ БАЗ ДАННЫХ НА ПЛАТФОРМЕ СУБД MYSQL С ИСПОЛЬЗОВАНИЕМ ТАБЛИЦ ТИПА MYISAM И INNODB (домашняя работа)

1 создание логической модели базы данных в виде ER-диаграммы согласно нотации IDEF1X;

2 создание и заполнение (для основных таблиц данными не менее 2000 записей) базы данных с таблицами типа MyIsam**.** Для каждой таблицы должны быть разработаны и определены:

а) SQL-запрос, определяющий тип таблицы;

б) первичный ключ (primary key);

в) индексные поля (foreign key), которые в дальнейшем будут использоваться для обеспечения целостности данных;

г) поля с аргументированным выбором типов данных и выделенными атрибутами «NOT NULL» и «NULL».

3 создание базы данных с таблицами типа InnoDB**.** Для каждой таблицы должны быть разработаны и определены:

а) SQL-запрос, определяющий тип таблицы;

б) первичный ключ (primary key); внешние ключевые поля (foreign key) для обеспечения связи с другими таблицами;

в) тип ссылочной целостности связи по внешнему ключу (ON UPDATE, CASCADE, SET NULL, NO ACTION, RESTRICT для инструкций UPDATE и DELETE);

г) поля с аргументированным выбором типов данных и выделенными атрибутами«NOT NULL» и «NULL».

1 РАЗРАБОТКА SQL-ЗАПРОСОВ С УЧЕТОМ СПЕЦИФИКИ ВЫСОКОНАГРУЖЕННЫХ БАЗ ДАННЫХ НА ПЛАТФОРМЕ СУБД MYSQL.

1 изучить особенности создания SQL-запросов для высоконагруженных систем хранения данных;

2 изучить синтаксис использования инструкции SELECT для создания SQL-запросов с различными условиями на выборку данных;

3 изучить синтаксис использования инструкций INSERT, UPDATE, DELETE для создания SQL-запросов манипулирования данными с заданными условиями;

4 изучить синтаксис использования инструкций GROUP BY (функций COUNT, AVG, MAX, MIN, конструкции WITHROLL UP) для создания SQL-запросов на выборку и группировку данных;

5 изучить синтаксис использования инструкций WHERE и JOIN (LEFT JOIN, RIGHT JOIN, INNER JOIN) для связывания данных различных таблиц в SQL-запросе;

6 изучить синтаксис использования инструкций LIKE, NOT LIKE, WHERE, HAVING, операторов NOT, ANY, ALL, SOME, IN, EXISTS для задания условий выполнения SQL-запроса;

7 изучить синтаксис создания сложных (вложенных один в другой) и составных (UNION) SQL-запросов;

8 изучить методику анализа плана выполнения SQL-запросов с помощью оператора EXPLAIN, назначение полей и сообщений таблицы EXPLAIN.

1 РЕАЛИЗАЦИЯ ПРОГРАММНОЙ ЛОГИКИ И ОБЕСПЕЧЕНИЕ ЦЕЛОСТНОСТИ СВЯЗЕЙ С ПОМОЩЬЮ ТРИГГЕРОВ ДЛЯ ВЫСОКОНАГРУЖЕННЫХ БАЗ ДАННЫХ НА ПЛАТФОРМЕ СУБД MYSQL

1 изучить определение, назначение и цели использования триггеров;

2 изучить синтаксис SQL-запроса CREATE TRIGGER для создания триггера;

3 изучить особенности создания триггера для условий его активации (BEFORE, AFTER);

4 изучить особенности создания триггера для всех возможных типов операций (INSERT, UPDATE, DELETE), активизирующих его выполнение;

5 изучить особенности использования предложений NEW.* и OLD.* в исполняемом SQL-коде триггера для различных условий и событий его активации;

6 изучить особенности использования операторов EXISTS (NOT EXISTS), IS NULL (IS NOT NULL) и функций IFNULL(), COALESCE(), NULLIF(), IF() COUNT(), для проверки условий существования записей;

7 изучить особенности отмены исполняемого SQL-кода триггера в зависимости от заданных условий и использования предложения SIGNAL;

8 изучить методику анализа плана выполнения SQL-запросов с помощью оператора EXPLAIN, назначение полей и сообщений таблицы EXPLAIN;

9 изучить особенности создания триггеров для высоконагруженных систем хранения данных.

3 СОЗДАНИЕ И ИСПОЛЬЗОВАНИЕ ПРЕДСТАВЛЕНИЙ ДЛЯ ВЫСОКОНАГРУЖЕННЫХ БАЗ ДАННЫХ НА ПЛАТФОРМЕ СУБД MYSQL.

1 изучить назначение, правила создания, хранения и использования представлений (VIEW) в СУБД MySQL (с существующими ограничениями);

2 алгоритмы MERGE и TEMPTABLE, используемые при работе с представлениями (VIEW);

3 возможные виды проверок корректности представлений (VIEW), поддерживаемые СУБД MySQL;

4 изучить синтаксис использования инструкции CREATE OR REPLACE VIEW для создания представлений (VIEW);

5 изучить особенности использования операторов INSERT, UPDATE, DELETE для манипулирования данными базовых таблиц с помощью представлений (VIEW);

6 изучить особенности использования SQL-запросов SELECT, когда в качестве источника данных используется представление (VIEW);

7 изучить особенности создания представлений (VIEW) для высоконагруженных систем хранения данных;

8 изучить методику анализа плана выполнения SQL-запросов к представлениям с помощью оператора EXPLAIN, назначение полей и сообщений таблицы EXPLAIN.

4 СОЗДАНИЕ ПРОЦЕДУР И ФУНКЦИЙ ДЛЯ ВЫСОКОНАГРУЖЕННЫХ БАЗ ДАННЫХ НА ПЛАТФОРМЕ СУБД MYSQL.

1 изучить назначение, правила создания, хранения и использования хранимых процедур (Stored Procedures) и функций (Stored Functions);

2 изучить синтаксис использования SQL-инструкции CREATE PROCEDURE для создания хранимых процедур (Stored Procedures);

3 изучить синтаксис использования SQL-инструкции CREATE FUNCTION для создания хранимых функций (Stored Functions);

4 изучить особенности использования операторов INSERT, UPDATE, DELETE для манипулирования данными базовых таблиц в коде хранимых процедур и функций;

5 изучить особенности использования функций CONCAT(), GROUP_CONCAT() для манипулирования форматом выборки информации с помощью инструкции SELECT;

6 изучить синтаксис использования локальных переменных, операторов управления потоком выполнения, в исполняемом SQL-коде хранимых процедур и функций (операторы IF, CASE, LOOP, REPEAT, WHILE, GOTO);

7 изучить особенности создания и использования курсоров (CURSORs) в хранимых процедурах;

8 изучить методику анализа плана выполнения SQL-запросов, использующихся в хранимых процедурах и функциях, с помощью оператора EXPLAIN. Назначение полей и сообщений таблицы EXPLAIN.

5 ОПТИМИЗАЦИЯ SQL-ЗАПРОСОВ ВЫСОКОНАГРУЖЕННЫХ БАЗ ДАННЫХ НА ПЛАТФОРМЕ СУБД MYSQL.

1 изучить особенности оптимизации запросов за счет применения таких понятий как:

– простые индексы;
– составные индексы;
– кластерные индексы;
– полнотекстовые индексы;

2 изучить понятие «селективность индекса» и научиться его использовать для оптимизации запросов;

3 изучить методику анализа плана выполнения SQL-запросов.

6 СОЗДАНИЕ ТРАНЗАКЦИЙ ДЛЯ ВЫСОКОНАГРУЖЕННЫХ БАЗ ДАННЫХ НА ПЛАТФОРМЕ СУБД MYSQL.

1 изучить определение, назначение и цели использования транзакций;

2 изучить виды блокировок, поддерживаемых СУБД MySQL для таблиц типа InnoDB и MyISAM;

3 изучить синтаксис использования инструкции START TRANSACTION для создания транзакций;

4 изучить особенности управления режимом автоматического подтверждения неявных транзакций с помощью динамической системной переменной СУБД MySQL @@autocommit.

5 изучить особенности использования SQL-операторов IF, COMMIT, ROLLBACK, синтаксис инструкций INSERT/REPLACE, UPDATE, DELETE для создания SQL-запросов манипулирования данными с заданными условиями;

6 изучить порядок, формат и состав данных фиксируемых в бинарном журнале СУБД MySQL при выполнении транзакций;

7 изучить особенности создания транзакций для высоконагруженных систем хранения данных.

Спасибо за внимание