

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Порохня Андрей Алексеевич
Должность: И.о. директора института перспективной инженерии
Дата подписания: 02.06.2026 12:53:32
Уникальный программный ключ:
990bea3aeec848c23bd8699e48288f8d1960c608

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

УТВЕРЖДАЮ

А.А. Порохня
Ф.И.О.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ
NoSQL СУБД и распределенная обработка данных
название дисциплины

Направление подготовки	09.04.03 Прикладная информатика
Направленность (профиль)	Управление знаниями
Год начала обучения	2026
Форма обучения	очная
Реализуется в семестре	3

Введение

1. Назначение: ФОС предназначен для объективной оценки уровня сформированности компетенций.

2. ФОС является приложением к программе дисциплины «NoSQL СУБД и распределенная обработка данных/ NoSQL DBMS and distributed data processing»

3. Разработчик Маликов А.В., профессор.

4. Проведена экспертиза ФОС.

Члены экспертной группы:

Председатель Азаров И.В., и.о. директора департамента

Члены комиссии:

Орлова А.Ю., доцент

Орлинская О.Г., доцент

Представитель организации-работодателя Черевко С.А., генеральный директор ООО "КРАФТКОД"

(Ф.И.О., должность)

Экспертное заключение: ФОС по дисциплине позволяет оценить уровень сформированности компетенций. Приступить к апробации.

5. Срок действия ФОС определяется сроком реализации образовательной программы.

1. Описание критериев оценивания компетенции на различных этапах их формирования, описание шкал оценивания

Уровни сформированности и компетенции(ий), индикатора (ов)	Дескрипторы			
	Минимальный уровень не достигнут (Неудовлетворительно) 2 балла	Минимальный уровень (удовлетворительно) 3 балла	Средний уровень (хорошо) 4 балла	Высокий уровень (отлично) 5 баллов
ПК-10				
ИД-1 ПК-10. развертывать, информационные системы, их компоненты и информационные сервисы	С трудом развертывает, информационные системы, их компоненты и информационные сервисы	Слабо развертывает, информационные системы, их компоненты и информационные сервисы	В целом развертывает, информационные системы, их компоненты и информационные сервисы	развертывает, информационные системы, их компоненты и информационные сервисы
ИД-2 ПК-10 инсталлировать информационные системы, их компоненты и информационные сервисы	С трудом инсталлирует информационные системы, их компоненты и информационные сервисы	Слабо инсталлирует информационные системы, их компоненты и информационные сервисы	В целом инсталлирует информационные системы, их компоненты и информационные сервисы	инсталлирует информационные системы, их компоненты и информационные сервисы
ПК-11				
ИД-1 ПК-11. гарантировать качество ИС	С трудом гарантирует качество ИС	Слабо гарантирует качество ИС	В целом гарантирует качество ИС	гарантирует качество ИС
ИД-2 ПК-11. гарантировать надежность ИС	С трудом гарантирует надежность ИС	Слабо гарантирует надежность ИС	В целом гарантирует надежность ИС	гарантирует надежность ИС.
ИД-3 ПК-11. гарантировать информационную безопасность ИС	С трудом гарантирует информационную безопасность ИС	Слабо гарантирует информационную безопасность ИС	В целом гарантирует информационную безопасность ИС	гарантирует информационную безопасность ИС
ПК-12				
ИД-1 ПК-12. использовать спецификации в профессиональной области	С трудом использует спецификации в профессиональной области	Слабо использует спецификации в профессиональной области	В целом использует спецификации в профессиональной области	использует спецификации в профессиональной области
ИД-2 ПК-12. использовать правила в профессиональной области	С трудом использует правила в профессиональной области	Слабо использует правила в профессиональной области	В целом использует правила в профессиональной области	использует правила в профессиональной области
ПК-6				
ИД-1 ПК-6. анализировать предметную область, идентифицировать и описывать проблемы, находить методы и подходы к их решению; формировать требования	С трудом анализирует предметную область, идентифицирует и описывает проблемы, находит методы и подходы к их решению; формирует требования	Слабо анализирует предметную область, идентифицирует и описывает проблемы, находит методы и подходы к их решению; формирует требования	В целом анализирует предметную область, идентифицирует и описывает проблемы, находит методы и подходы к их решению; формирует требования	анализирует предметную область, идентифицирует и описывает проблемы, находит методы и подходы к их решению; формирует требования

ИД-2 ПК-6. Идентифицировать, классифицировать и описывать проблемы предметной области, находить методы и подходы к их решению.	С трудом идентифицирует, классифицирует и описывает проблемы предметной области, находит методы и подходы к их решению.	Слабо идентифицирует, классифицирует и описывает проблемы предметной области, находит методы и подходы к их решению.	В целом идентифицирует, классифицирует и описывает проблемы предметной области, находит методы и подходы к их решению.	идентифицирует, классифицирует и описывает проблемы предметной области, находит методы и подходы к их решению.
ПК-8				
ИД-1 ПК-8. профессионально эксплуатировать ИКТ и вычислительного оборудования для информатизации прикладных задач.	С трудом профессионально эксплуатирует ИКТ и вычислительное оборудование для информатизации прикладных задач	Слабо профессионально эксплуатирует ИКТ и вычислительное оборудование для информатизации прикладных задач	В целом профессионально эксплуатирует ИКТ и вычислительное оборудование для информатизации прикладных задач	профессионально эксплуатирует ИКТ и вычислительное оборудование для информатизации прикладных задач
ИД-2 ПК-8. профессионально проводить маркетинговый анализ ИКТ и вычислительного оборудования для информатизации прикладных задач.	С трудом проводит маркетинговый анализ ИКТ и вычислительного оборудования для информатизации прикладных задач	Слабо проводит маркетинговый анализ ИКТ и вычислительного оборудования для информатизации прикладных задач	В целом проводит маркетинговый анализ ИКТ и вычислительного оборудования для информатизации прикладных задач	профессионально проводит маркетинговый анализ ИКТ и вычислительного оборудования для информатизации прикладных задач
ПК-9				
ИД-1 ПК-9. разрабатывать информационные системы, их компоненты и информационные сервисы	С трудом разрабатывает информационные системы, их компоненты и информационные сервисы	Слабо разрабатывает информационные системы, их компоненты и информационные сервисы	В целом разрабатывает информационные системы, их компоненты и информационные сервисы	разрабатывает информационные системы, их компоненты и информационные сервисы
ИД-2 ПК-9. интегрировать информационные системы, их компоненты и информационные сервисы	С трудом интегрирует информационные системы, их компоненты и информационные сервисы	Слабо интегрирует информационные системы, их компоненты и информационные сервисы	В целом интегрирует информационные системы, их компоненты и информационные сервисы	интегрирует информационные системы, их компоненты и информационные сервисы

Оценивание уровня сформированности компетенции по дисциплине осуществляется на основе «Положения о проведении текущего контроля успеваемости и промежуточной аттестации обучающихся по образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры - в федеральном государственном автономном образовательном учреждении высшего образования «Северо-Кавказский федеральный университет» в актуальной редакции.

ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ПРОВЕРКИ УРОВНЯ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ

Номер задания	Правильный ответ	Содержание вопроса	Компетенция
Форма обучения очная, семестр 3			
1.	Язык. Реляционные базы данных используют структурированный язык запросов (Structured Query Language, SQL) для определения и обработки данных. Нереляционные базы данных, в свою очередь, предлагают динамическую структуру данных, которые могут храниться несколькими способами: ориентированно по колонкам, документо-ориентированно, в виде графов или на основе пар «ключ-значение». Такая гибкость означает следующее: Вы можете создавать документы, не задавая их структуру заранее; Каждый документ может обладать собственной структурой; У каждой базы данных может быть собственный синтаксис; Вы можете добавлять поля прямо во время работы с данными. Масштабируемость В большинстве случаев SQL базы данных вертикально масштабируемые, то есть вы можете увеличивать нагрузку на отдельно взятый сервер, наращивая мощность центральных процессоров, объёмы ОЗУ или системы хранения данных. А NoSQL базы данных горизонтально масштабируемы. Это означает, что вы можете увеличивать трафик, распределяя его или добавляя больше серверов к вашей СУБД. Структура В реляционных СУБД данные представлены в виде таблиц, в то время как в нереляционных — в виде документов, пар «ключ-значение», графов или wide-column хранилищ. Это делает SQL базы данных лучшим выбором для приложений, которые предполагают транзакции с несколькими записями — как,	Основные различия NoSQL и РСУБД	ПК-6 ПК-8

	например, система учётных записей — или для устаревших систем, которые были построены для реляционных структур.		
2.	Всего есть 4 основных типа: Хранилище «ключ-значение» — в нём есть большая хеш-таблица, содержащая ключи и значения. Примеры: Riak, Amazon DynamoDB; Документоориентированное хранилище — хранит документы, состоящие из тегированных элементов. Пример: CouchDB; Колоночное хранилище — в каждом блоке хранятся данные только из одной колонки. Примеры: HBase, Cassandra; Хранилище на основе графов — сетевая база данных, которая использует узлы и рёбра для отображения и хранения данных. Пример: Neo4J.	Назовите типы хранилищ NoSQL СУБД	ПК-6 ПК-8
3.	Отсутствие схемы в базах данных «ключ-значение», например, Riak, — это как раз то, что вам нужно для хранения данных. Ключ может быть синтетическим или автосгенерированным, а значение может быть представлено строкой, JSON, блобом (BLOB, Binary Large Object, большой двоичный объект) и т.д. Такие базы данных как правило используют хеш-таблицу, в которой находится уникальный ключ и указатель на конкретный объект данных. Существует понятие блока (bucket) — логической группы ключей, которые не группируют данные физически. В разных блоках могут быть идентичные ключи. И хотя базы данных типа «ключ-значение» могут пригодиться в определённых ситуациях, они не лишены недостатков. Первый заключается в том, что модель не предоставляет стандартные возможности баз данных вроде атомарности транзакций или согласованности данных при одновременном выполнении нескольких транзакций. Такие возможности должны предоставляться самим приложением. Второй недостаток в том, что при увеличении объёмов данных, поддержание уникальных ключей может стать проблемой. Для её решения необходимо как-то усложнять процесс генерации	База данных типа «ключ-значение»	ПК-6 ПК-8

	строк, чтобы они оставались уникальными среди очень большого набора ключей. Riak и Dynamo от Amazon — самые популярные СУБД данных такого типа.		
4.	Данные, представленные парами ключ-значение, сжимаются как хранилище документов схожим с хранилищем «ключ-значение» образом, с той лишь разницей, что хранимые значения (документы) имеют определённую структуру и кодировку данных. XML, JSON и BSON — некоторые из стандартных распространённых кодировок. <pre>{officeName: "3Pillar Noida", {Street: "B-25", City: "Noida", State: "UP", Pincode: "201301"}} }</pre> <pre>{officeName: "3Pillar Timisoara", {Boulevard: "Coriolan Brediceanu No. 10", Block: "B, Ist Floor", City: "Timisoara", Pincode: "300011"}} }</pre> <pre>{officeName: "3Pillar Cluj", {Latitude: "40.748328", Longitude: "-73.985560"}} }</pre> Одним из ключевых различий между хранилищем «ключ-значение» и документоориентированным является то, что последний включает метаданные, связанные с хранимым содержимым, что даёт возможность делать запросы на основе содержимого. Например, в указанном примере можно попробовать найти все документы, в которых «City» равно «Noida», что вернёт все документы, связанные с магазинами в этом городе. Apache CouchDB — пример документоориентированной СУБД. CouchDB использует JSON для хранения данных, JavaScript в	Документоориентированная база данных	ПК-6 ПК-8

	качестве языка запросов с использованием MapReduce и HTTP для API. Данные и отношения не хранятся в таблицах так, как в традиционных реляционных базах данных, а по сути являются набором независимых документов. Тот факт, что такие базы данных работают без схемы, делает простой задачей добавление полей в JSON-документы без необходимости сначала заявлять об изменениях. Couchbase и MongoDB — самые популярные документоориентированные СУБД.		
5.	В колоночных NoSQL базах данных данные хранятся в ячейках, сгруппированных в колонки, а не в строки данных. Колонки логически группируются в колоночные семейства. Колоночные семейства могут состоять из практически неограниченного количества колонок, которые могут создаваться во время работы программы или во время определения схемы. Чтение и запись происходит с использованием колонок, а не строк. В сравнении с хранением данных в строках, как в большинстве реляционных баз данных, преимущества хранения в колонках заключаются в быстром поиске/доступе и агрегации данных. Реляционные базы данных хранят каждую строку как непрерывную запись на диске. Разные строки хранятся в разных местах на диске, в то время как колоночные базы данных хранят все ячейки, относящиеся к колонке, как непрерывную запись, что делает операции поиска/доступа быстрее. Пример: получение списка заголовков нескольких миллионов статей будет трудоёмкой задачей при использовании реляционных баз данных, так как для извлечения заголовков придётся проходить по каждой записи. А можно получить все заголовки с помощью только одной операции доступа к диску. Модель данных Колоночное семейство — структура, которая может легко группировать колонки и суперколонки;	Колоночная база данных	ПК-6 ПК-8

	Ключ — постоянное имя записи. У ключей может быть разное количество колонок, поэтому база данных может расширяться неравномерно; Пространство ключей — определяет самый внешний уровень организации, как правило, имя приложения/базы данных. Колонка — имеет упорядоченный список элементов — кортежей с именами и значениями. Самыми известными примерами являются Google BigTable и HBase с Cassandra, вдохновлённые BigTable.		
6.	В графовой базе данных вы не найдёте строгого формата SQL или представления таблиц и колонок, вместо этого используется гибкое графическое представление, которое идеально подходит для решения проблем масштабируемости. Графовые структуры используются вместе с рёбрами, узлами и свойствами, что обеспечивает безиндексную смежность. При использовании графового хранилища данные могут быть легко преобразованы из одной модели в другую. <pre> graph TD G[Граф] -- Записи --> U[Узлы] G -- Записи --> O[Отношения] U -- Имеют --> S[Свойства] O -- Имеют --> S U <--> Упорядочивают O </pre> Такие базы данных используют рёбра и узлы для представления данных. Узлы связаны между собой определёнными отношениями, представленными рёбрами между ними. У узлов и отношений есть некоторые свойства. Контейнерная иерархия документоориентированной базы данных содержит данные без схемы, которые можно представить в виде дерева, которое является графом. Если обращаться к документам или их элементам в этом дереве, можно получить более выразительное представление данных, в котором можно легко ориентироваться с помощью Neo4j	Графовая база данных	ПК-6 ПК-8

7.	Динамическая схема: Как упоминалось выше, эта СУБД позволяет гибко работать со схемой данных без необходимости изменять сами данные; Масштабируемость: MongoDB горизонтально масштабируема, что позволяет легко уменьшить нагрузку на сервера при больших объёмах данных; Удобство в управлении: СУБД не нуждается в отдельном администраторе базы данных. Благодаря достаточному удобству в использовании, ей легко могут пользоваться как разработчики, так и системные администраторы; Скорость: Высокая производительность при выполнении простых запросов; Гибкость: В MongoDB можно без вреда для существующих данных, их структуры и производительности СУБД добавлять поля или колонки.	MongoDB: нереляционная СУБД. Преимущества MongoDB:	ПК-6 ПК-8
8.	Термин NoSQL образован от англ.: «Not Only SQL» – «Не только SQL»	Что означает термин NoSQL?	ПК-9 ПК-10
9.	Концепция NoSQL предоставляет высокую доступность и устойчивость данных к разделению, но при этом в NoSQL не все хорошо с согласованностью данных. Этот подход призван решать собственный класс задач.	Какие преимущества предоставляют NoSQL базы данных в сравнении с реляционными базами данных?	ПК-9 ПК-10
10.	Основными особенностями MongoDB являются:	Какими особенностями обладает MongoDB?	ПК-11

	• документно-ориентированное хранилище; • полная поддержка индексов; • репликация данных; • высокая доступность данных; • способность к горизонтальному масштабированию; • авто-шардинг; • поддержка запросов; • поддержка Map/Reduce; • поддержка GridFS .		ПК-12
11.	Mongo может работать в двух режимах: • в качестве обычного приложения Windows. Запуск из командной строки. • в качестве Windows-сервиса. Автоматически запускается при старте системы.	Сколькими способами можно произвести установку MongoDB? Кратко опишите эти способы.	ПК-9 ПК-10
12.	☐ приложения ядра базы данных, ☐ инструменты для дампа бинарных файлов MongoDB, ☐ инструменты импорта и экспорта данных, ☐ инструменты для диагностики, ☐ приложения распределенной файловой системы.	На какие группы делятся приложения, входящие в состав MongoDB?	ПК-11 ПК-12
13.	Набор логически связанных между собой совокупностей разделяемых данных (и их описаний), которые физически распределены в некоторой компьютерной сети	Распределенная база данных	ПК-11 ПК-12
14.	Программный комплекс, предназначенный для управления распределенными базами данных и обеспечивающий прозрачный доступ пользователей к распределенной информации.	Распределенная СУБД	
15.	Имеется набор логически связанных разделяемых данных. Сохраняемые данные разбиты на некоторое количество фрагментов. Может быть предусмотрена репликация фрагментов данных. Фрагменты и их копии распределяются по разным узлам. Узлы связаны между собой сетевыми соединениями.	Характеристики распределённой СУБД	ПК-11 ПК-12

	Доступ к данным на каждом узле происходит под управлением СУБД. СУБД на каждом узле способна поддерживать автономную работу локальных приложений. СУБД каждого узла поддерживает хотя бы одно глобальное приложение		
16.	Обработка с использованием централизованной базы данных, доступ к которой может осуществляться с различных компьютеров сети. Ключевым моментом в определении распределенной СУБД является утверждение, что система работает с данными, физически распределенными в сети. Если данные хранятся централизованно, то даже в том случае, когда доступ к ним обеспечивается для любого пользователя по сети, эта система просто поддерживает распределенную обработку, но не может рассматриваться как распределенная СУБД.	Распределенная обработка	ПК-11 ПК-12
17.	Система управления базой данных, функционирующей с использованием нескольких процессоров и жестких дисков, что позволяет ей (если это возможно) распараллеливать выполнение некоторых операций с целью повышения общей производительности обработки.	Параллельная СУБД	ПК-9 ПК-10
18.	системы с разделением памяти; системы с разделением дисков; системы без разделения вычислительных ресурсов.	Типа архитектуры параллельных СУБД	ПК-9 ПК-10
19.	состоят из тесно связанных между собой компонентов, в число которых входит несколько процессоров, разделяющих общую системную память. Эта архитектура, называемая также архитектурой с симметричной многопроцессорной обработкой (SMP), в настоящее время получила широкое распространение и применяется для самых разных вычислительных платформ, от персональных рабочих станций, содержащих несколько параллельно работающих микропроцессоров, больших RISC-систем и вплоть до крупнейших мэйнфреймов. Эта архитектура	Системы с разделением памяти	ПК-9 ПК-10

	обеспечивает быстрый доступ к данным для ограниченного набора процессоров, количество которых обычно не превосходит 64. В противном случае взаимодействие по сети становится узким местом всей системы.		
20.	создаются из менее тесно связанных между собой компонентов. Они являются оптимальным вариантом для приложений, которые унаследовали высокую централизацию обработки и должны обеспечивать самые высокие показатели доступности и производительности. Каждый из процессоров имеет непосредственный доступ ко всем совместно используемым дисковым устройствам, но обладает собственной оперативной памятью. Как и в случае архитектуры без разделения вычислительных ресурсов, архитектура с разделением дисков исключает узкие места, связанные с совместно используемой памятью. Однако, в отличие от архитектуры без разделения вычислительных ресурсов, данная архитектура исключает упомянутые узкие места без внесения дополнительных издержек, связанных с физическим распределением данных по отдельным устройствам. Разделяемые дисковые системы иногда называют кластерами.	Системы с разделением дисков	ПК-9 ПК-10
21.	(эту архитектуру иначе называют архитектурой с массовой параллельной обработкой) используют схему, в которой каждый процессор, являющийся частью системы, имеет свою собственную оперативную и дисковую память. База данных распределена между всеми дисковыми устройствами, подключенным к отдельным, связанным с этой базой данных вычислительным подсистемам, в результате чего все данные прозрачно доступны пользователям каждой из этих подсистем. Такая архитектура обеспечивает более высокий уровень масштабируемости, чем системы с разделяемой памятью, и позволяет легко поддерживать большое количество процессоров. Однако оптимальной производительности удастся	Системы без разделения вычислительных ресурсов	ПК-9 ПК-10

	достичь только в том случае, если требуемые данные хранятся локально.		
22.	– документно-ориентированное хранилище; – полная поддержка индексов; – репликация данных; – высокая доступность данных; – способность к горизонтальному масштабированию; – авто-шардинг; – поддержка запросов; – поддержка Map/Reduce; – поддержка GridFS6 .	Основные особенности MongoDB	ПК-9 ПК-10
23.	это тиражирование изменений данных с главного сервера БД на одном или нескольких зависимых серверах	Репликация	ПК-9 ПК-10
24.	разделение данных на уровне ресурсов, разбиение данных по какому-либо признаку. Концепция шардинга заключается в логическом разделении данных по различным ресурсам.	Шардинг	ПК-9 ПК-10
25.	☐ По умолчанию MongoDB создает базу данных с именем «test»	Создает ли MongoDB по умолчанию какую-либо базу данных? Если создает, назовите её имя.	ПК-9 ПК-10
26.	reduce map	Какие процессы составляют MapReduce? a. insert b. reduce c. map	ПК-9 ПК-10 ПК-11 ПК-12
27.	Массив	Какую структуру данных присваивают оператору \$or? a. Объект b. Строку c. Массив d. Число	ПК-9 ПК-10 ПК-11 ПК-12
28.	пропуск первых 5 документов коллекции и возврат всех остальные документы	Результатом выполнения операции db.unicorns.find().skip(5) будет a. возврат первых 5 документов b. возврат последних 5 документов c. возврат всех документов, содержащих цифру	ПК-9 ПК-10 ПК-11 ПК-12

		5 d. пропуск первых 5 документов коллекции и возврат всех остальных документов	
29.	отсутствие установки и сведённые к минимуму расхождения сООП гибкость	Какие преимущества предоставляет бесструктурность документ-ориентированных баз данных? a. отсутствие установки и сведённые к минимуму расхождения сООП b. гибкость c. хаотичная организация данных	ПК-9 ПК-10 ПК-11 ПК-12
30.	коллекцию и _id документа, на который он ссылается	DBRef включает в себя a. коллекцию и имя документа, на который он ссылается b. коллекцию и _id документа, на который он ссылается c. все документы в коллекции d. все связанные документы	ПК-9 ПК-10 ПК-11 ПК-12
31.	allPlansExecution	Какой режим будет использовать метод курсора explain(), если передать ему в качестве параметра verbose значение true? a. executionStats b. allPlansExecution c. не имеет режимов d. queryPlanner	ПК-9 ПК-10 ПК-11 ПК-12
32.	mongodump	Какая утилита позволяет выполнять операции резервного копирования в MongoDB? a. mongorestore b. mongoimport c. mongoexport d. mongodump	ПК-9 ПК-10 ПК-11 ПК-12
33.	нет	Поддерживает ли MongoDB транзакции? a. да b. нет	ПК-9 ПК-10 ПК-11

		с. не все	ПК-12
34.	\$or	В MongoDB логическое ИЛИ реализовано оператором? а. or б. \$or в. г. ИЛИ	ПК-9 ПК-10 ПК-11 ПК-12
35.	шардинг	Для хранения частей базы данных на разных серверах применяется: а. шардинг б. денормализация в. репликация г. унификация	ПК-9 ПК-10 ПК-11 ПК-12
36.	4 Мб	Какое ограничение по объёму имеет одиночный документ: а. 10 Мб б. 1 Мб в. 2 Мб г. 4 Мб	ПК-9 ПК-10 ПК-11 ПК-12
37.	ошибку	При выборке в коллекции users запрос db.users.find({}, {name:0, age:1, gender:0, _id:1}) вернет: а. документы без полей name и gender б. документы с полями age и _id в. документы без полей name, gender и с полями age, _id г. ошибку	ПК-9 ПК-10 ПК-11 ПК-12
38.	2	Сколько этапов имеет MapReduce-подход? а. 1 б. 3 в. 2 г. 4	ПК-9 ПК-10 ПК-11 ПК-12

39.	наборы фиксированного размера, которые поддерживают высокую пропускную способность операций, которые вставляют и извлекают документы, основанные на порядке вставки	Ограниченная коллекция — это a. коллекция с ограниченным количеством документов b. наборы фиксированного размера, которые поддерживают высокую пропускную способность операций, которые вставляют и извлекают документы, основанные на порядке вставки c. коллекция, накладывающая определенные ограничения на поля документов d. коллекция, в которой запрещены определенные операции с данными	ПК-9 ПК-10 ПК-11 ПК-12
40.	может быть различной может быть одинаковой	Структура документов коллекции должна быть различной a. может быть различной b. может быть одинаковой c. должна быть одинаковой	ПК-9 ПК-10 ПК-11 ПК-12
41.	тело метода	Что будет возвращено в консоль при вызове методов exit, help или других без скобок на конце (т.е. не так help(), а так help)? a. голова метода b. тело метода c. ошибка	ПК-9 ПК-10 ПК-11 ПК-12
42.	system.indexes	В какой коллекции можно узнать информацию обо всех индексах базы данных? a. Indexes b. students c. unicorns d. system.indexes	ПК-9 ПК-10 ПК-11 ПК-12
43.	будут отсортированы по возрастанию	При следующем запросе db.users.find({}).sort({age:-1, age:1}) в конечном результате документы: a. будут отсортированы по возрастанию b. не будут отсортированы c. будут отсортированы по убыванию	ПК-9 ПК-10 ПК-11 ПК-12

44.	не ограничена	Глубина вложенности документов: a. не ограничена b. ограничена 2-мя уровнями вложенности c. ограничена 1-им уровнем вложенности d. ограничена 3-мя уровнями вложенности	ПК-9 ПК-10 ПК-11 ПК-12
45.	вложенные документы и технология DBRef	Для агрегации данных в случаях отношения «один-ко-многим» или «многие-ко-многим» классические СУРБД применяют конструкцию JOIN. Для таких случаев в MongoDB используется a. аналог конструкции JOIN b. только вложенные документы c. только технология DBRef d. вложенные документы и технология DBRef	ПК-9 ПК-10 ПК-11 ПК-12
46.	reduce	На каком этапе emit-пары являются входными данными? a. reduce b. emit-пары — исключительно выходные данные c. map	ПК-9 ПК-10 ПК-11 ПК-12
47.	создание нового документа и связать его с документом, имеющим ObjectId(«4d85c7039ab0fd70a117d731»)	Результатом выполнения команды db.employees.insert({_id: ObjectId(«4d85c7039ab0fd70a117d731»), name: 'Duncan', manager: ObjectId(«4d85c7039ab0fd70a117d730»)}); будет a. создание нового документа b. в коде ошибка c. создание нового документа и связать его с документом, имеющим ObjectId(«4d85c7039ab0fd70a117d731»)	ПК-9 ПК-10 ПК-11 ПК-12
48.	копирование всех документов из одной коллекции в другую	Результатом выполнения команды db.source.copyTo(target) будет a. копирование всех документов из одной коллекции в другую	ПК-9 ПК-10 ПК-11 ПК-12

		b. копирование всех полей из одного документа в другой c. копирование определенных полей документа d. копирование всех коллекций	
49.	удаляет указанный индекс из коллекции	Метод dropIndex a. изменяет указанный индекс b. добавляет указанный индекс c. ищет указанный индекс в коллекции d. удаляет указанный индекс из коллекции	ПК-9 ПК-10 ПК-11 ПК-12
50.	5	Сколько параметров имеет метод update согласно документации Mongo DB? a. 4 b. 2 c. 5 d. 3 e. 1	ПК-9 ПК-10 ПК-11 ПК-12