# Билет №1 - сделано

В файле *Задача 2. Число образовательных организаций и численность студентов в Приволжском ФО.xlsx* имеются данные по регионам Приволжского федерального округа о числе образовательных организаций в 2020 г. и численности студентов на начало учебного года.

Изображение выглядит как текст, снимок экрана, линия, число

Автоматически созданное описание

Требуется создать краткий отчет и настроить визуализации данных в разрезе регионов федерального округа в Microsoft Power BI.

# Билет №2 - сделано

В *файле Задача 1. Численность студентов на начало уч. года.xlsx* имеются данные Росстата по регионам Российской Федерации на начало учебного года.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Требуется создать краткий отчет и настроить визуализации данных в разрезе Российской Федерации, федеральных округов и регионов в Yandex DataLens.

Ход выполнения:

1. Создать подключение файл с данными и геоточки (Задача Численность студентов)
2. Создать датасет, связь по региону, настройка Поля (среднее и геоточка) (Числ-ть студентов)
3. Создать чарты по РФ, ФО и регионам.
4. Ссылка на дашборд: <https://datalens.yandex.cloud/we7yjn40uoaok-chislennost-studentov>

# Билет №3 – написать запросы

Используя учебную базу данных и кроссплатформенный менеджер DBeaver, выполните запросы, которые позволят определить, какова цена заказа на каждую дату. Отсортируйте по id заказа. Выведите id заказа, дату заказа и цену единицы товара.

Выполните два варианта запроса: без использования и с использованием оператора inner join. Убедитесь, что предложенные вами два запроса дают идентичные результаты.

Этот SQL запрос выполняет выборку данных из таблиц "orders" и "order\_details" для получения информации о заказах. Здесь используется оператор JOIN для объединения строк в двух таблицах по условию равенства столбца "order\_id". Запрос агрегирует данные по номеру заказа и дате заказа, а также вычисляет общую стоимость каждого заказа с учетом скидок. Результат сгруппирован по номеру заказа и отсортирован по номеру заказа.

без оператора inner join

select o.order\_id, o.order\_date as дата\_заказа, sum(od.unit\_price\*quantity - (od.unit\_price\*quantity)\*discount) as цена\_заказа from orders o, order\_details od where od.order\_id = o.order\_id group by o.order\_id, o.order\_date

с оператором: Этот запрос извлекает данные из таблицы `orders` и `order\_details`, объединяет их по `order\_id` и затем группирует результаты по `order\_id` и `order\_date`. Для каждого заказа вычисляется общая цена заказа с учетом цены за единицу товара, количества и скидки на товары. Затем результаты сортируются по `order\_id` в порядке возрастания.

SELECT o.order\_id, o.order\_date AS дата\_заказа,

SUM(od.unit\_price \* od.quantity - (od.unit\_price \* od.quantity) \* od.discount) AS цена\_заказа

FROM orders o

INNER JOIN order\_details od ON o.order\_id = od.order\_id

GROUP BY o.order\_id, o.order\_date

ORDER BY o.order\_id;

# Билет №4

Выполните Задание 1.2 в соответствии с условиями, представленными по ссылке:

<https://github.com/BosenkoTM/MySQL_TASKS/blob/main/TASKS/Task1.md>

Текст задания:

Задание 1.2 Крупные страны

Страна является крупной страной, если ее площадь составляет более 3 миллионов квадратных километров, или (с населением более 25 миллионов). Создайте таблицу World и вставьте следующие строки данных.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Указания.

1.2.1 Создайте таблицу World.

1.2.2 Вставьте данные в таблицу World.

1.2.3 Напишите SQL-запрос, который выводит в таблицу **названия**, **население** и **площади всех крупных стран.**

Требуемый результат выполнения DML- и DDL-скриптов в SQL Fiddle:

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Сервис: <https://sqlfiddle.com/>

КОД:

CREATE TABLE World (

name VARCHAR(255),

continent VARCHAR(255),

area INT,

population BIGINT);

INSERT INTO World (name, continent, area, population) VALUES

('Afghanistan', 'Asia', 652230, 25500100),

('Albania', 'Europe', 28748, 2831741),

('China', 'Asia', 9598962, 1412600000),

('France', 'Europe', 547030, 67841000),

('Russia', 'Europe', 17125191, 145478097);

SELECT \* FROM World;

SELECT name, population, area

FROM World

WHERE area > 3000000 OR population > 25000000;

# Билет №5 сппр – проверить

База: СППР Модульный экзамен

Используя «1С:Система проектирования прикладных решений», зафиксируйте группирующую функцию «Оптовые продажи клиентам» и распишите в ней следующие Функции системы:

* Прием заказа клиента;
* Выставление счета клиенту;
* Фиксация оплаты по счету;
* Подбор товара на складе;
* Реализация и доставка товара клиенту.

Схема группирующей функции:

Изображение выглядит как текст, снимок экрана, диаграмма, Прямоугольник

Автоматически созданное описание

Для каждой Функции системы проработайте состав объектов Метаданных.

Требования к программной архитектуре:

* Должны быть представлены следующие объекты: Справочники, Документы, Отчеты, Регистры накоплений (Регистры сведений – по необходимости);
* Для каждого объекта должен быть представлен пример реквизитного состава, включая реквизиты табличных частей;
* Для регистров накоплений определите Измерения и Ресурсы, Измерения должны иметь ссылочный тип данных;
* Для ряда реквизитов определите ссылочный тип данных.

Продемонстрируйте результаты в виде ряда ER-диаграмм.

**На первом этапе** выполняется сбор сведений об автоматизируемых процессах предприятия. Определяются последовательность и условия выполнения работ, альтернативные и смежные работы, исходные данные для начала работ и условия их завершения, должностные лица, задействованные в производственном процессе. Процессы описываются в СППР.

Целью **второго этапа** (логического проектирования) является определение функциональности и объема данных, которыми должна оперировать разрабатываемая информационная система. На этом этапе выполняется формализация процессов и представление их в виде функций, выполняемых пользователями в разрабатываемой конфигурации, и информационных потоков.

Результатом **этапа логического проектирования** является согласованная с процессами логическая модель разрабатываемой конфигурации.

Логическая модель включает в себя:

■ Функции разрабатываемой конфигурации, представленные на схемах IDEF0 с заданной степенью детализации.

■ Информационные потоки, связывающие функции между собой и с внешним окружением разрабатываемой информационной системы.

■ Профили пользователей, являющиеся исполняющими связями на схемах IDEF0 и определяющие единичные функциональные обязанности пользователей и их права в информационной системе.

В основу разрабатываемой логической модели положена методология IDEF0. Согласование логической модели выполняется по ряду критериев, описанных в главе, посвященной логическому проектированию. Логическая модель системы может трактоваться как техническое задание для проектирования.

На **третьем этапе** производится разработка архитектуры информационной системы.

На этом этапе данные логической модели детализируются до объектов метаданных: справочников, документов, планов видов характеристик, констант и т. п. Устанавливается соответствие между объектами метаданных и функциями в СППР, что позволяет оценить роль и место объектов метаданных в логической модели системы, проанализировать полноту и правильность проектных решений.

Подсистемы разрабатываемой конфигурации и их иерархия достаточно просто выделяются из полученной на этапе логического проектирования декомпозиции функций. Подсистема может включать в себя одну или несколько функций.

При необходимости на этапе разработки архитектуры системы может уточняться логическая модель информационной системы.

Результатами третьего этапа проектирования являются метаданные разрабатываемой конфигурации, согласованные с логической моделью.

# Билет №6.

Практика.

Выполнить операции по аналитике данных в экосистеме cloudera. Провести обработку больших данных на основе [Pig](https://github.com/BosenkoTM/cloudera-quickstart/blob/main/README.md#pig):

* В интерактивном режиме через Терминал, запустить оболочку Pig;
* В Hue перейти в редактор Pig через Query > Editor > Pig;
* Запустите следующий скрипт/команды, чтобы загрузить и отобразить первые десять строк из файла геолокации:

geoloc = LOAD 'geoloc/geolocation.csv' USING PigStorage(',') AS (truckid,driverid,event,latitude,longitude,city,state,velocity,event\_ind,idling\_ind);

geoloc\_limit = LIMIT geoloc 10;

DUMP geoloc\_limit;

* Проверить папку «results», хранящуюся в HDFS, по строке STORE result. Пояснить полученные результаты;
* Проанализировать журналы на новой вкладке Hadoop > YARN Resource Manager в Firefox;
* Подсчитать список различных городов, посещенных каждым идентификатором грузовика, и среднюю скорость для каждого идентификатора грузовика.

Постановка задачи представлена по ссылке: <https://github.com/BosenkoTM/cloudera-quickstart/blob/main/README.md>

Решение.

Общие моменты от Босенко Т.М.: <https://github.com/BosenkoTM/cloudera-quickstart/blob/main/README.md>

Конкретно наша задача: https://github.com/BosenkoTM/cloudera-quickstart/blob/main/hw\_3-2\_hdfs\_pig.md

Скачать архив с виртуалкой по ссылке: https://downloads.cloudera.com/demo\_vm/virtualbox/cloudera-quickstart-vm-5.13.0-0-virtualbox.zip

Разархивировать, в папке будет 2 файла:



Далее открываем программу Virtual Box.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Выберите в качестве конфигурации этот файл:



Установить для виртуальной машины диапазон виртуальной памяти от 4 до 8 ГБ ОЗУ (у меня более 4000 МБ, это около 4 ГБ). Плюс я увеличила количество процессоров с 1 до 2, если можете больше, делайте больше, но не отдавайте все процессоры компа (у меня всего 4).

PS: оказалось, что это очень мало, машина работала очень медленно.

Изображение выглядит как текст, снимок экрана, программное обеспечение, веб-страница

Автоматически созданное описание

Далее запустите машину.

Чтобы растянуть окно машины можете нажать сюда (чтобы отменить надо нажать правый ctrl + C):

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Чтобы включить копипасту, включите двунаправленный буфер обмена:

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

☹ Машина очень медленная, не спешите по 10 раз жмякать, надо ждать после каждого клика.

Откройте терминал:

Изображение выглядит как текст, программное обеспечение, компьютер, Операционная система

Автоматически созданное описание

Установим разрешения на запись пользователем hdfs. Запустите следующий скрипт.

sudo su -

su hdfs

hdfs dfs -chmod -R 777 /tmp

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Откройте новый терминал и введите команду для открытия браузера:

firefox

Изображение выглядит как текст, Шрифт, снимок экрана, число

Автоматически созданное описание

В браузере перейдите на закладку Heu. Учетные данные: cloudera/cloudera.

Изображение выглядит как текст, снимок экрана, Шрифт, программное обеспечение

Автоматически созданное описание

Создать каталог ex\_3\_2:

mkdir ex\_3\_2

Пререйти в каталог ex\_3\_2:

cd ex\_3\_2

Изображение выглядит как текст, Шрифт, белый, типография

Автоматически созданное описание

Скачать данные Geolocation data:

wget https://community.cloudera.com/legacyfs/online/attachments/2768-geolocation.zip

Изображение выглядит как текст, чек, Шрифт, снимок экрана

Автоматически созданное описание

разархивировать данные

unzip 2768-geolocation.zip

Изображение выглядит как текст, чек, Шрифт, белый

Автоматически созданное описание

В Hue, выбрать Browsers > Files

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Создайте новый каталог в HDFS с именем geoloc/ внутри HDFS из Hue

Изображение выглядит как снимок экрана, текст, программное обеспечение, Шрифт

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Заходим в папку

Изображение выглядит как текст, Шрифт, линия, число

Автоматически созданное описание

Загрузите Geolocation.csv и trucks.csv в только что созданную папку geoloc/.

Изображение выглядит как текст, снимок экрана, линия, Шрифт

Автоматически созданное описание

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание

Зажимая shift, можете выбрать сразу оба файла, open.

Изображение выглядит как текст, программное обеспечение, Шрифт, Значок на компьютере

Автоматически созданное описание

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

В Hue перейти в редактор Pig через Query > Editor > Pig

Изображение выглядит как текст, Шрифт, число, линия

Автоматически созданное описание

Если прям очень долго грузится, обновите (еще раз нажмите на Hue).

Если выходит такая штука, нажимайте Continue (у меня вылезала постонно).

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

Сначала убедиться в запуске Dataflow инструмента Oozie.

Через + добавьте в браузере новую вкладку и выберите Oozie.

Изображение выглядит как текст, Шрифт, программное обеспечение, снимок экрана

Автоматически созданное описание

У меня там пока так:

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Возвращаемся на вкладку с Hue и вставляем этот скрипт (это скрипт из gita):

geoloc = LOAD 'geoloc/geolocation.csv' USING PigStorage(',') AS (truckid:chararray,

driverid:chararray, event:chararray, latitude:double, longitude:double, city:chararray,

state:chararray, velocity:double, event\_ind:long, idling\_ind:long);

geoloc\_limit = LIMIT geoloc 10;

STORE geoloc\_limit INTO 'results-geoloc';

DUMP geoloc\_limit;

И запускаем его:

Изображение выглядит как текст, Шрифт, линия, число

Автоматически созданное описание

В самом билете у нас немного другой скрипт:

geoloc = LOAD 'geoloc/geolocation.csv' USING PigStorage(',') AS

(truckid,driverid,event,latitude,longitude,city,state,velocity,event\_ind,idling\_ind);

geoloc\_limit = LIMIT geoloc 10;

DUMP geoloc\_limit;

GPT говорит, что разница между ними следующая: «В первом скрипте после загрузки данных из файла geolocation.csv они также выводятся на экран с помощью DUMP, но затем ограниченное количество строк (10) сохраняется в файл 'results-geoloc'. Во втором скрипте после загрузки данных из файла geolocation.csv они остаются в памяти Pig и выводятся на экран с помощью DUMP. Данные не сохраняются нигде еще».

Поскольку по заданию мы должны будем просмотреть результат в папке, нам подходит скрипт из гита, а не из билета.

А тем временем, если перейти на вкладку с Oozie и обновить там, мы увидим:

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Если нажать по Джобу 2 раза и перейти в Job Dag, должен быть его рисунок.

Изображение выглядит как текст, снимок экрана, диаграмма, программное обеспечение

Автоматически созданное описание

Вернемся к файлам (см. стр. 6). Там появилась новая директория, зайдем в нее:

Изображение выглядит как текст, снимок экрана, число, линия

Автоматически созданное описание

Далее идем сюда:

Изображение выглядит как текст, снимок экрана, число, линия

Автоматически созданное описание

А вот и результат:

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

Нам надо пояснить этот результат. Вот, как помог GPT:

“Данный скрипт Pig загружает данные из файла geolocation.csv в отношении геолокации транспортных средств. Затем устанавливает ограничение на количество строк (10 строк) и сохраняет полученные данные в файл 'results-geoloc'. Наконец, выводит первые 10 строк данных.

Полученные данные представляют собой информацию о различных транспортных средствах, таких как идентификатор грузовика (truckid), идентификатор водителя (driverid), тип события (event), широту и долготу местоположения (latitude и longitude), город (city), штат (state), скорость (velocity), индикатор события (event\_ind) и индикатор простоя (idling\_ind).

Например, первая строка показывает данные о грузовике с идентификатором A19 и водителем A19, который находится в городе San Pablo, штат Калифорния, с нулевой скоростью и показателем события 0 (нормальное событие) и показателем простоя 1.

Аналогично, остальные строки представляют информацию о различных грузовиках, их расположении, скорости и событиях.”

Проанализировать журналы на новой вкладке Hadoop > YARN Resource Manager в Firefox. Для этого через + создайте новую вкладку в браузере, а там:

Изображение выглядит как текст, программное обеспечение, Значок на компьютере, веб-страница

Автоматически созданное описание

Там будет:

Изображение выглядит как текст, снимок экрана, число, программное обеспечение

Автоматически созданное описание

«В разделе Cluster Metrics отображается количество приложений, которые были отправлены, ожидают выполнения, выполняются и завершены, количество работающих контейнеров, используемая оперативная память, общий объем оперативной памяти, зарезервированная оперативная память, используемые ядра процессора, общее количество ядер процессора и зарезервированные ядра процессора.

В разделе Cluster Nodes Metrics показано количество активных узлов кластера, узлов, которые находятся в стадии вывода из эксплуатации, узлов, которые были выведены из эксплуатации, узлов, которые были потеряны, узлов, на которых есть проблемы, и узлов, которые были перезагружены.

В разделе User Metrics для пользователя с именем "dr.who" показана информация о приложениях, отправленных им, ожидающих выполнения, выполняющихся, завершившихся, работающих контейнерах, контейнерах, ожидающих выполнения, зарезервированных контейнерах, используемой оперативной памяти, оперативной памяти, ожидающей выполнения, зарезервированной оперативной памяти, используемых ядрах процессора, ядрах процессора, ожидающих выполнения и зарезервированных ядрах процессора.

Далее представлены конкретные приложения с их характеристиками, такими как ID, имя, тип приложения, очередь запуска, время начала, время завершения, состояние выполнения, окончательный статус, кол-во работающих контейнеров, выделенные виртуальные ядра, выделенная память MB, зарезервированные ядра процессора, зарезервированная память и прогресс.»

КОРОЧЕ, видим выполненные вычисления и что они завершены успешно.

И последняя задачка:

Подсчитать список различных городов, посещенных каждым идентификатором грузовика, и среднюю скорость для каждого идентификатора грузовика.

Алгоритм решения:

1. Сгруппировать данные по идентификатору грузовика (truckid) с помощью оператора GROUP BY и использовать встроенную функцию Pig COUNT для подсчета уникальных городов и AVG для расчета средней скорости.

2. Использовать оператор FOREACH для создания новой структуры данных, содержащей идентификатор грузовика, количество уникальных городов и среднюю скорость.

3. Сохранить результат с помощью оператора STORE в отдельный файл.

geoloc = LOAD 'geoloc/geolocation.csv' USING PigStorage(',') AS (truckid:chararray,

driverid:chararray, event:chararray, latitude:double, longitude:double, city:chararray,

state:chararray, velocity:double, event\_ind:long, idling\_ind:long);

grouped\_geoloc = GROUP geoloc BY truckid;

result = FOREACH grouped\_geoloc {

unique\_cities = DISTINCT geoloc.city;

avg\_speed = AVG(geoloc.velocity);

GENERATE group AS truckid, COUNT(unique\_cities) AS num\_cities\_visited, avg\_speed AS average\_speed;

}

STORE result INTO 'results-geoloc1';

Копируем и вставляем скрипт, который выше. Если не вставляется, то вставлять частями.

И запускаем:

Изображение выглядит как текст, снимок экрана, программное обеспечение, число

Автоматически созданное описание

Заглянем в Oozie:

Изображение выглядит как текст, снимок экрана, Шрифт, программное обеспечение

Автоматически созданное описание

Изображение выглядит как снимок экрана, текст, диаграмма, программное обеспечение

Автоматически созданное описание

В папке появилась новая директория:

Изображение выглядит как текст, снимок экрана, программное обеспечение, число

Автоматически созданное описание

Зайдем в нее и далее сюда:

Изображение выглядит как текст, снимок экрана, число, программное обеспечение

Автоматически созданное описание

Результат:

Изображение выглядит как текст, снимок экрана, программное обеспечение, число

Автоматически созданное описание

1-ый столбец – идентификатор грузовика, 2-ой – количество посещенных городов, 3-ий – средняя скорость.

Чтобы выключить машину, наберите команду в терминале:

sudo shutdown now

**ПОЛЕЗНЫЕ ТЕРМИНЫ**

**Hadoop** – это свободно распространяемый набор утилит, библиотек и фреймворк для разработки и выполнения распределённых программ, работающих на кластерах из сотен и тысяч узлов. Эта основополагающая технология хранения и обработки больших данных ( Big Data ) является проектом верхнего уровня фонда Apache Software Foundation.

**Cloudera** — это платформа программного обеспечения для обработки больших данных, которую выбирают представители многих отраслей. Она предоставляет клиентам такие компоненты, как Hadoop, Spark и Hive.

**Apache Pig** — это платформа, позволяющая создавать программы для Hadoop с помощью процедурного языка, известного как Pig Latin. Apache Pig создает более простую абстракцию процедурного языка над платформой MapReduce, чтобы реализовать SQL-подобный интерфейс для приложений Hadoop.

**Apache Oozie** — это серверная система планирования рабочих процессов для управления заданиями Hadoop.

**YARN** – это фреймворк управления ресурсами в Apache Spark, другими словами, это почти операционная система на кластерном уровне.

# Билет №7 сппр -проверить

СППР для разработки

Используя «1С:Система проектирования прикладных решений» для выбранного предприятия, выполните классификацию процессов, сгруппируйте процессы по следующим пяти категориям:

* Базовые процессы;
* Ограничивающие процессы;
* Обеспечивающие процессы;
* Процессы производства;
* Процесса управление развитием.

Для выбранных трех процессов из разных групп выполните их описание:

* Укажите краткое Описание процесса;
* Определите, когда стартует и чем завершается процесс;
* Зафиксируйте Шаги процесса, для каждого дайте краткое Описание;
* Приведите примеры формулировки Требований к системе для некоторых Шагов процесса.

**Базовые процессы** – процессы реализации требований клиентов, включая операции приема заказа клиентов, обслуживания клиентов, ведения взаиморасчетов, выполнения складских и транспортных операций. Базовые бизнес-процессы генерируют доходы компании. К ним относятся процессы, ориентированные на производство товара или оказание услуги, являющиеся целевыми объектами создания предприятия и обеспечивающие получение дохода. Именно основные бизнес-процессы формируют результат и потребительские качества, за которые внешний клиент готов платить деньги.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Ограничивающие процессы - процессы разработки ограничений деятельности компании. Наличие в операциях базового процесса (и других процессов) какого-либо документа, регламентирующего его выполнение или поведение субъекта.

Изображение выглядит как текст, снимок экрана, Шрифт

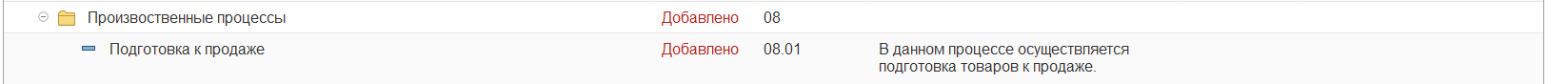
Автоматически созданное описание

Обеспечивающие процессы – процессы обеспечения организации различного рода ресурсами

Изображение выглядит как текст, Шрифт, число, снимок экрана

Автоматически созданное описание

Производственные процессы компании – все производственные, технологические процессы



Процессы управления развитием – процессы, связанные с совершенствованием бизнес-процессов, определении точек контроля в процессе, разработке и реализации системы сбалансированных показателей, контроллинге показателей, управлении изменениями, связи стратегического управления с операционным

Изображение выглядит как текст, Шрифт, линия, снимок экрана

Автоматически созданное описание

# Билет №8 сппр проверть

СППР для разработки

Используя «1С:Система проектирования прикладных решений» для выбранного предприятия, выполните описание Функций системы.

Функции системы сгруппируйте по следующим категориям:

* Базовые процессы;
* Ограничивающие процессы;
* Обеспечивающие процессы;
* Процессы производства;
* Процесса управление развитием.

В группе Базового процесса и Ограничивающего процесса зафиксируйте как минимум по одному процессу (группирующую Функцию) и для выбранных двух группирующих Функций распишите Функции системы (должно быть представлено минимум по 5 функций для каждой группирующей функции).

Требования к описанию Функций:

* Укажите наименование и краткое описание функции, а также сформулируйте результат функции;
* Для функции укажите Входящие данные и Потребителей, где применимо;
* Продемонстрируйте варианты Видов связей для Входящих данных: Вход и Управление;
* Заведите Роли и укажите их в Исполнителях к функции.

Продемонстрируйте Схемы полученных функций, начиная с Группирующей функции и далее переходя по всем Функциям системы.

Изображение выглядит как текст, снимок экрана, Шрифт, диаграмма

Автоматически созданное описание

# Билет №9.

Теория по Элме:

Элма. «Заявка на приобретение техники».

Пример из методички:

Изображение выглядит как текст, диаграмма, снимок экрана, число

Автоматически созданное описание

Рис. 93. Графическая модель бизнес-процесса " Заявка на приобретение техники"

# Билет №10 показатели фролов сделать

Спроектируйте в таблице показатели компании Ланит для перспективы «Управление персоналом» (https://job.lanit.ru/Pages/default.aspx). Укажите цель компании по данному направлению и не менее трех возможных инициатив для ее достижения, а также измеримые показатели для каждой инициативы и формулы для их расчета.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Инициатива | Показатель | Формула для расчета показателя | Целевое значение показателя | | | Ответственный в компании за инициативу (должность) |
| неуд | удовл. | хорошо |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

Проведите защиту своей концепции планирования деятельности компании.

BSC – это система, позволяющая анализировать связи между между финансовыми показателями, внутренними процессами, ситуацией на рынке, интересами и моральным состоянием сотрудников.

Стратегические процессы: 1. Оценка стратегических результатов деятельности организации Таблицы BSC (score – счет, рейтинг, card – карточка) содержат информацию в трех формах: 1) текстовая (миссия, видение, цели, мероприятия, индикаторы для измерения достижения стратегических целей), 2) графическая – стратегия в виде дерева стратегических целей (подцелей), 3) числовая – значения ключевых показателей. Используется 4 группы показателей: 1) финансы, 2) клиенты, 3) бизнес-процессы (технологии), 4) персонал (рост и обучение). Показателям может быть придана роль лидирующих индикаторов, выполняющих функцию тревожного сигнала. Лаговые индикаторы отражают прошлые тенденции. Таким образом, система BSC выполняет роль системы раннего оповещения о негативных тенденциях в развитии компании.

Функции Balanced Scorecard • BSC – это декларация. Стратегия изображается и документируется в форме, понятной владельцам бизнеса, его инвесторам, персоналу. • BSC – это план. Стратегия трансформируется в оперативную деятельность. • BSC – это коммуникатор. На стадии разработки BSC в компании формируется общее представление о долгосрочном развитии компании. • BSC – это контролер. Реализация стратегии контролируется по ключевым показателям с минимальным запаздыванием.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, диаграмма, мультфильм

Автоматически созданное описание

Проблемы Balanced Scorecard • Традиционная BSC не в полной мере балансирует интересы групп влияния, поскольку следует «обычной» логике бизнеса: компания должна обогащать владельцев, эксплуатируя клиентов и персонал. При таком подходе финансовый результат становится скорее целью, нежели средством. • Концепции «общества рядовых сотрудников (The Employee Society)», «обучающейся организации (The Learning Organization)» отражают фундаментальные перемены, происходящие в постиндустриальном обществе. «Капиталистический» подход к приоритетам существования бизнеса все более (по мнению упомянутых авторов) ухудшает долгосрочные перспективы компаний. • Отчасти примирить BSC и новые законы ведения экономических дел возможно через модель «Баланс интересов», предлагающей иной подход к взаимоотношениям групп влияния и компании. По нему финансовая перспектива BSC «смещается» на нижний уровень иерархии и служит «источником финансирования процесса удовлетворения групп влияния компании». При сохранении общей логики построения BSC, существенно изменяется направленность бизнеса в целом

Изображение выглядит как текст, снимок экрана, диаграмма, мультфильм

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, число, Параллельный

Автоматически созданное описание

Примечание: Полученные сертификаты сами по себе не гарантируют повышение эффективности работника и, как следствие, компании в целом. Таким образом, доля сертификатов является дисфункциональным показателем, нежелательным к использованию Проектная команда должна задуматься о том, как мотивировать персонал на повышение эффективности своей деятельности и на применение полученных по итогам обучения знаний, навыков в операционной деятельности. S серт sales – число сотрудников, сертифицированных по технике продаж S серт proc – число сотрудников, сертифицированных по отбору продукции.

Цель: Создание эффективной и устойчивой кадровой стратегии

Эта цель объединяет все аспекты: Сокращение времени найма и снижение текучести кадров: Обучение и улучшение условий работы для привлечения и удержания лучших специалистов. Увеличение производительности: Инвестирование в обучение, что позволит сотрудникам улучшить свои навыки и знания, что в свою очередь повысит производительность. Повышение уровня удовлетворенности сотрудников: Оптимизация рабочей среды, мотивация и удовлетворенность персонала благоприятно скажется на их продуктивности и loyality. Такая кадровая стратегия поможет создать привлекательную и поддерживающую среду для сотрудников, что в итоге способствует успеху и процветанию компании

Изображение выглядит как текст, снимок экрана, число, Параллельный

Автоматически созданное описание

Обеспечить стабильность, эффективность и высокую производительность персонала через оптимизацию процессов набора, удержания, обучения и повышения удовлетворенности сотрудников. – Цель

https://miro.com/app/board/uXjVKAAyHEo=/

# Билет №11.

Элма «Обработка заявок службы технической поддержки»

Изображение выглядит как текст, снимок экрана, диаграмма, Шрифт

Автоматически созданное описание

Рис. 94. Графическая модель бизнес-процесса "Обработка заявок службы технической поддержки"

# Билет 12.

Элма. «Прием на рабтоу нового сотрудника»

Изображение выглядит как текст, снимок экрана, диаграмма, Параллельный

Автоматически созданное описание

Рис. 95."Прием на работу нового сотрудника"

# Билет 13.

Элма. «Согласование бюджета»

Buhgalter\_4

Podrazdel\_1

Podrazdel\_2

Изображение выглядит как текст, диаграмма, снимок экрана, линия

Автоматически созданное описание

Рис. 96. Графическая модель бизнес-процесса "Согласование бюджета"

# Билет №14 – hdfs проверить

Задание. Примените внешние таблицы при работе с большими данными.

1. Скачать <https://datasets.imdbws.com/name.basics.tsv.gz>
2. Создать каталог HDFS /user/hadoop/imdb/name\_basics/ для файла name.basics.tsv
3. Создайте внешнюю Hive таблицу name\_basics для name.basics.tsv
4. Используйте HiveQL, чтобы ответить на следующие вопросы:
   1. Сколько фильмов и сериалов находится в наборе данных IMDB?
   2. Кто самый молодой актер/сценарист/... в наборе данных?
   3. Создайте список (tconst, original\_title, start\_year, average\_rating, num\_votes), который состоит из:
      1. фильм вышел в 2010 году или позднее;
      2. фильм имеет средний рейтинг, равный или превышающий 8,1;
      3. проголосовали более 100 000 раз.
   4. Сколько фильмов находится в списке c)

Постановка задачи: <https://github.com/BosenkoTM/BigDataAnalitic_Practice/tree/main/exercises/winter_semester_2021-2022/02_hive>

Примерный алгоритм решения задачи: <https://github.com/BosenkoTM/BigDataAnalitic_Practice/blob/main/solutions/winter_semester_2021-2022/02_hive/Exercise_2.pdf>

Виртуальная машина «Ubuntu».

Алгоритм в файлах часть\_1, часть\_2.

Выполнение пункта 4.

a.Сколько фильмов и сериалов находится в наборе данных IMDB?

SELECT m.title\_type, count(\*) FROM title\_basics m GROUP BY m.title\_type;

b.Кто самый молодой актер/сценарист/... в наборе данных?

Вариант поиска именно актер/сценарист/...

SELECT \* FROM name\_basics

Where birth\_year is not null

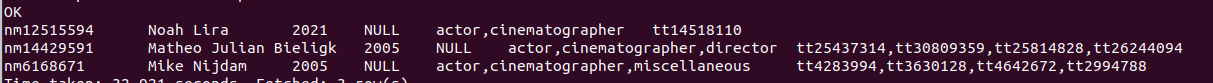
And primary\_professional like ‘%actor,cinematographer%’

Order by bith\_year desc

Limit 5;

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание



Либо просто самый молодой

SELECT \* FROM name\_basics n WHERE n.birth\_year = ( SELECT MAX(birth\_year) FROM name\_basics);



Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

c.Создайте список (tconst, original\_title, start\_year, average\_rating, num\_votes), который состоит из: фильм вышел в 2010 году или позднее; фильм имеет средний рейтинг, равный или превышающий 8,1; проголосовали более 100 000 раз.

SELECT m.tconst, m.original\_title, m.start\_year, r.average\_rating, r.num\_votes FROM title\_basics m

JOIN title\_ratings r on (m.tconst = r.tconst)

WHERE r.average\_rating >= 8.1

and m.start\_year >= 2010

and m.title\_type = 'movie'

and r.num\_votes > 100000

ORDER BY r.average\_rating desc, r.num\_votes DESC;

d.Сколько фильмов находится в списке c)

SELECT count(\*) FROM title\_basics m

JOIN title\_ratings r on (m.tconst = r.tconst)

WHERE r.average\_rating >= 8.1 and m.start\_year >= 2010 and m.title\_type = 'movie' and r.num\_votes > 100000;

# Билет №15 файл колаб описать спарк

# Билет №16. Сайт скл

Выполните Задание 1.1 в соответствии с условиями, представленными по ссылке: <https://github.com/BosenkoTM/MySQL_TASKS/blob/main/TASKS/Task1.md>

Текст задания:

Найти повторяющиеся электронные письма. Создайте таблицу email и вставьте следующие строки данных.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Указания.

1.1.1 Создайте таблицу email.

1.1.2 Вставьте данные в таблицу email.

1.1.3 Напишите SQL-запрос, чтобы найти все повторяющиеся адреса электронной почты в таблице email.

Требуемый результат выполнения DML- и DDL-скриптов в SQL Fiddle:

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

КОД:

-- Создание таблицы

CREATE TABLE email (

ID INT AUTO\_INCREMENT KEY,

Email VARCHAR(25)

);

-- Заполнение таблицы

INSERT INTO email(Email)

VALUES

('a@b.com'),

('c@d.com'),

('a@b.com'),

('с@d.com'),

('b@a.com'),

('b@a.com'),

('pipe@mgpu.ru'),

('mgtu@bmstu.ru'),

('pipe@mgpu.ru');

-- Вывод таблицы

SELECT \* FROM email;

-- Запрос для всех повторяющихся адресов эл. почты

SELECT email

FROM email

GROUP BY Email

HAVING COUNT(\*) > 1

# Билет №17.сайт скл

Выполните Задание 3.1 в соответствии с условиями, представленными по ссылке:

<https://github.com/BosenkoTM/MySQL_TASKS/blob/main/TASKS/Task3.md>

Задание 3.1 Самые высокооплачиваемые сотрудники в каждом отделе. Создайте таблицу Employee, которая содержит всю информацию о сотрудниках, у каждого сотрудника есть соответствующий идентификатор Id, зарплата salary и идентификатор отдела department Id.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Указания.

3.1.1 Создайте таблицу Employee.

3.1.2 Вставьте данные в таблицу courses.

3.1.3 Создайте таблицу Department с информацией обо всех отделах компании.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

3.1.4 Вставьте данные в таблицу Department.

3.1.5 Напишите SQL-запрос, чтобы найти самого высокооплачиваемого сотрудника в каждом отделе.

Требуемый результат выполнения DML- и DDL-скриптов в SQL Fiddle:

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

КОД:

CREATE TABLE Employee (

Id INT AUTO\_INCREMENT KEY,

Name VARCHAR(20),

Salary INT,

DepartmentId INT

);

INSERT INTO Employee(Name, Salary, DepartmentId) VALUES ('Joe', 70000, 1);

INSERT INTO Employee(Name, Salary, DepartmentId) VALUES ('Henry',80000 , 2);

INSERT INTO Employee(Name, Salary, DepartmentId) VALUES ('Sam', 60000,2 );

INSERT INTO Employee(Name, Salary, DepartmentId) VALUES ('Max',90000 ,1 );

SELECT \* FROM Employee;

CREATE TABLE Department (

Id INT AUTO\_INCREMENT KEY,

Name VARCHAR(20)

);

INSERT INTO Department(Name) VALUES ('IT');

INSERT INTO Department(Name) VALUES ('Sales');

SELECT \* FROM Department;

SELECT Department, Name as Employee, Salary

from (SELECT d.name as Department, DepartmentId, e.Name, Salary, ROW\_NUMBER() OVER (PARTITION BY

DepartmentId ORDER BY Salary DESC) as SalaryRank

FROM Employee e join Department d ON d.Id = e.DepartmentId) as Ranked

where Ranked.SalaryRank=1

ORDER BY DepartmentId;

# Билет №18.сайт скл

Выполните Задание 2.1 в соответствии с условиями, представленными по ссылке:

<https://github.com/BosenkoTM/MySQL_TASKS/blob/main/TASKS/Task2.md>

Текст задания:

Задание 2.1 Обучение с детьми

Определить класс, в котором обучается более 5 учеников. Создайте таблицу courses, как показано ниже, со следующими элементами: student и class

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Указания.

2.1.1 Создайте таблицу courses.

2.1.2 Вставьте данные в таблицу courses.

2.1.3 Напишите запрос SQL, в котором перечисляются все классы, где обучается более 5 учеников.

Note：Учащиеся не должны учитываться дважды в одном и том же классе.

Требуемый результат выполнения DML- и DDL-скриптов в SQL Fiddle:

Изображение выглядит как текст, Шрифт, снимок экрана, белый

Автоматически созданное описание

КОД:

CREATE TABLE courses (

student VARCHAR(100),

class VARCHAR(100)

);

INSERT INTO courses (student, class) VALUES

('A', 'Math'),

('B', 'English'),

('C', 'Math'),

('D', 'Biology'),

('E', 'Math'),

('F', 'Computer'),

('G', 'Math'),

('H', 'Math'),

('I', 'Math'),

('A', 'Math');

SELECT \* from courses;

SELECT class

FROM (

SELECT DISTINCT student, class

FROM courses

) AS unique\_courses

GROUP BY class

HAVING COUNT(student) > 5;

# Билет 19 сайт скл

Выполните Задание 4.1 в соответствии с условиями, представленными по ссылке:

<https://github.com/BosenkoTM/MySQL_TASKS/blob/main/TASKS/Task4.md>

Текст задания:

Задание 4.1 Такси и клиенты

В таблице Trips хранится информация о поездках в такси. Каждая поездка имеет уникальный идентификатор Id. Client\_Id и Driver\_Id являются внешними ключами для Users\_Id в таблице Users. Status — это тип перечисления, а членами перечисления являются (completed, cancelled\_by\_driver, cancelled\_by\_client).

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

В таблице Users хранятся все клиенты. У каждого клиента есть свой уникальный ключ Users\_Id. Banned указывает, забанен клиент или нет, а Role представляет собой тип перечисления (client, driver, partner).

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

4.1.1 Создайте таблицу Users.

4.1.2 Создайте таблицу Employee.

4.1.3 Вставьте данные в таблицу Users.

4.1.4 Вставьте данные в таблицу Trips.

Напишите SQL-запрос, чтобы узнать количество отмен для незаблокированных пользователей (Cancellation Rate) с 1 октября 2021 года по 3 октября 2021 года. Cancellation Rate округляется до двух знаков после запятой.

Требуемый результат выполнения DML- и DDL-скриптов в SQL Fiddle:

КОД:

CREATE TABLE Users (

Users\_Id INT PRIMARY KEY,

Banned ENUM('Yes', 'No'),

Role ENUM('client', 'driver', 'partner')

);

CREATE TABLE Trips (

Id INT PRIMARY KEY,

Client\_Id INT,

Driver\_Id INT,

City\_Id INT,

Status ENUM('completed', 'cancelled\_by\_driver', 'cancelled\_by\_client'),

Request\_at DATE

);

INSERT INTO Users (Users\_Id, Banned, Role) VALUES

(1, 'No', 'client'),

(2, 'Yes', 'client'),

(3, 'No', 'client'),

(4, 'No', 'client'),

(10, 'No', 'driver'),

(11, 'No', 'driver'),

(12, 'No', 'driver'),

(13, 'No', 'driver');

INSERT INTO Trips (Id, Client\_Id, Driver\_Id, City\_Id, Status, Request\_at)

VALUES

(1, 1, 10, 1, 'completed', '2021-10-01'),

(2, 2, 11, 1, 'cancelled\_by\_driver', '2021-10-01'),

(3, 3, 12, 6, 'completed', '2021-10-01'),

(4, 4, 13, 6, 'cancelled\_by\_client', '2021-10-01'),

(5, 1, 10, 1, 'completed', '2021-10-02'),

(6, 2, 11, 6, 'completed', '2021-10-02'),

(7, 3, 12, 6, 'completed', '2021-10-02'),

(8, 2, 12, 12, 'completed', '2021-10-03'),

(9, 3, 10, 12, 'completed', '2021-10-03'),

(10, 4, 13, 12, 'cancelled\_by\_driver', '2021-10-03');

SELECT

Request\_at AS Day,

ROUND(

SUM(CASE WHEN Status LIKE 'cancelled%' AND U.Banned = 'No' THEN 1 ELSE 0 END) /

NULLIF(SUM(CASE WHEN U.Banned = 'No' THEN 1 ELSE 0 END), 0),

2) AS `Cancellation Rate`

FROM Trips T

JOIN Users U ON T.Client\_Id = U.Users\_Id

WHERE Request\_at BETWEEN '2021-10-01' AND '2021-10-03'

GROUP BY Request\_at;

# Дополнительно

Мое задание