Оленчикова Т.Ю.

# Лабораторная работа 4.

# Hadoop. Разработка распределенных программ

**Цель:** Ознакомиться фремворком Apache Spark для реализации распределённой обработки неструктурированных и слабоструктурированных данных, входящий в экосистему проектов Hadoop.

### Порядок выполнения работы:

Напишите программу на Python, используя фреймворк Apache Spark, согласно вашему варианту для датасета “brooklyn\_sales\_map.csv”. Для помощи в выполнении задания можно воспользоваться книгами «Изучаем Spark. Молниеносный анализ данных -2015», «PySpark SQL Recipes» или интернетом.

### Индивидуальные задания

**Вариант 1.**

* Найдите среднюю стоимость жилья (sale\_price) и выведите новую таблицу, содержащую две колонки – стоимость жилья и отклонение стоимости от среднего значения.
* Найдите среднюю стоимость жилья (sale\_price) для каждого района.
* Выведите среднюю полную площадь жилья (gross\_sqft) для всех сочетаний налоговых категорий (tax\_class) и лет продажи (year\_of\_sale).
* Выведите таблицу, содержащую количество пустых (null) значений для каждой колонки.

**Вариант 2.**

* Найдите среднюю площадь жилья (gross\_sqft) и выведите новую таблицу, содержащую стоимость жилья и отклонение стоимости от среднего значения.
* Найдите среднюю площадь жилья (gross\_sqft) для каждого года, в котором оно было построено (year\_built).
* Найдите среднюю стоимость жилья (sale\_price) для всех сочетаний соседств (neighborhood) и категорий класса здания (building\_class\_category).
* В исходном датафрейме удалите все строчки с записями домов, которые были построены позже 2000 года (year\_built), а также те, которые содержат только нулевые значения.

**Вариант 3.**

* Найдите среднюю стоимость жилья (sale\_price) и выведите новую таблицу, содержащую две колонки – стоимость жилья и процент отклонения стоимости от среднего значения.
* Выведите таблицу, содержащую все категории класса зданий (building\_class\_category) и количество записей, которые к ним относятся.
* Выведите таблицу, содержащую средние значения по каждому столбцу в датафрейме.
* В исходном датафрейме заполните все нулевые значения средними по столбцу.

**Вариант 4.**

* Добавьте в датасет новую колонку, содержащую «возраст» жилья.
* Выведите таблицу, содержащую среднюю дату продажи для всех сочетаний индексов (zip\_code) и налоговых категорий (tax\_class) жилья.
* Выведите таблицу, содержащую суммарную стоимость жилья (sale\_price) по всем сочетаниям налоговых категорий (tax\_class) и индексов (zip\_code).
* Создайте новую таблицу, в которой есть 10 колонок исходного датафрейма, в которых нулевые значения есть, но не преобладают. После этого удалите все строки, в которых содержатся исключительно нули.

**Вариант 5.**

* Найдите средний год постройки жилья (year\_built) и выведите новую таблицу, содержащую год постройки жилья и отклонение года постройки от среднего значения.
* Отсортировать датасет по возрастанию цены продажи (sale\_price) и убыванию индексов (zip\_code) одновременно.
* Выведите таблицу с наибольшими ценами продажи (sale\_price) и количеством зданий по каждому сочетанию соседства (neighborhood) и категории класса здания (building\_class\_category

### Требования к оформлению отчета.

Отчет по ЛР состоит из

**а)**Титульный лист: название ЛР, ФИО студента и номер группы, № варианта.

б)Текст задания; в) UML-диаграмму классов (создаете в Umlet):

г) Исходный текст программы;

д) Скрины результатов выполнения;

е) Выводы.

### Критерии оценивания.

Оценка суммируется из следующих оценок:

1. Приложение отлажено и работоспособно -2 балла; имеются ошибки или реализованы не все функции – 1 балл; приложение не работоспособно – 0 баллов

2. Отчет оформлен в соответствии с требованиями – 1 балл, иначе – 0 баллов

3. Полные и верные ответы на контрольные вопросы – 1 балл, иначе – 0 баллов

Итого – максимум 4 балла.

Внимание! Полученная оценка автоматически снижается на 2% за каждую полную неделю задержки сдачи отчета по работе, но не более, чем на 40%.

### Контрольные вопросы

1. Что такое архитектура Apache Spark?
2. Что вы понимаете под трансформациями в Spark?
3. Как реализована потоковая передача в Spark?
4. Проиллюстрируйте некоторые недостатки использования Spark.
5. Объясните пять V больших данных
6. Как работает Spark Streaming?
7. Как реализована потоковая передача в Spark?
8. Какие существуют типы преобразования в DStream?
9. Hadoop использует репликацию для обеспечения отказоустойчивости. Как это достигается в Apache Spark?
10. Расскажите об основных функциях Apache Spark.
11. Что такое наборы данных Spark?
12. Что такое YARN в Spark?
13. Каковы преимущества Spark по сравнению с MapReduce?
14. Есть ли польза от изучения MapReduce, если Spark лучше, чем MapReduce?
15. Что такое перетасовка в Spark? Когда это происходит?
16. Что такое приемники в Apache Spark Streaming?
17. Предоставляет ли Apache Spark контрольные точки?
18. Что такое скалярные и агрегатные функции в Spark SQL?
19. Зачем нужны широковещательные переменные при работе с Apache Spark?
20. Перечислите несколько варианто в использования, в которых Spark превосходит Hadoop в обработке.
21. Какие языки поддерживает Apache Spark для разработки приложений, выполняющих обработку больших данных?

### Литература

1. Изучаем Spark: молниеносный анализ данных / Х. Карау, Э. Конвински, П. Венделл, М. Захария. — Москва : ДМК Пресс, 2015. — 304 с.— URL: https://e.lanbook.com/book/90118 (дата обращения: 15.09.2023). — Режим доступа: для авториз. пользователей.
2. Бутаков, Н. А. Обработка больших данных с Apache Spark : учебно-методическое пособие / Н. А. Бутаков, М. В. Петров, Д. Насонов. — Санкт-Петербург : НИУ ИТМО, 2019. — 50 с. — URL: https://e.lanbook.com/book/136573 (дата обращения: 15.09.2023). — Режим доступа: для авториз. пользователей.
3. Перрен, Ж. -. Spark в действии / Ж. -. Перрен ; перевод с английского А. В. Снастина. — Москва : ДМК Пресс, 2021. — 636 с.— URL: https://e.lanbook.com/book/241001 (дата обращения: 15.09.2023). — Режим доступа: для авториз. пользователей.
4. % https://www.currentschoolnews.com/ru/%D1%81%D1%82%D0%B0%D1%82%D1%8C%D0%B8/%D0%B7%D0%B0%D0%B6%D0%B5%D1%87%D1%8C-%D0%B2%D0%BE%D0%BF%D1%80%D0%BE%D1%81%D1%8B-%D0%BD%D0%B0-%D1%81%D0%BE%D0%B1%D0%B5%D1%81%D0%B5%D0%B4%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B8/D0%B0%D1%82%D1%8C%D0%B8/%D0%B7%D0%B0%D0%B6%D0%B5%D1%87%D1%8C-%D0%B2%D0%BE%D0%BF%D1%80%D0%BE%D1%81%D1%8B-%D0%BD%D0%B0-%D1%81%D0%BE%D0%B1%D0%B5%D1%81%D0%B5%D0%B4%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B8/