Cloud Computing

Lightning-fast unified analytics engine

Übungsbeispiele zur Klausurvorbereitung

Cloud Computing, 6. Semester Bachelor: Wirtschaftsinformatik

FH CAMPUS 02
Informationstechnologien & Wirtschaftsinformatik

Grahsl Hans-Peter

1 Basket Analyzer mit Spark SQL und DSL

Sie bekommen einen kleinen Auszug an Warenkorb-Daten und werden mittels SPARK SQL einfache Auswertungen für die Einkäufe machen.

1) Sie finden im package *edu.campus02.iwi.prfg* bereits die **Klasse BasketAnalyzer** in der Sie eine Spark Configuration samt Spark Session definieren sollen. *[1 Punkt]*

2) **Lesen Sie die Datei basketinfos.json** aus dem Ordner *data/input/* in ein Dataframe d.h. Dataset<Row> ein. *[1 Punkt]*

3) **Registrieren Sie ein View** mit dem namen „baskets“, welches Sie später mittels SQL abfragen werden. *[1 Punkt]*

4) Erzeugen Sie ein neues Dataframe indem Sie die originalen, eingelesenen Daten filtern, sodass nur mehr jene Baskets enthalten sind, welche nicht mit „MasterCard“ bezahlt wurden, welche aus den Kategorien „Toys“ oder „Music“ sind und zwischen 150.00 und 350.00 (jeweils inkl.) an Bestellwert aufweisen. Verwenden Sie dafür direkt **SQL und NICHT(!) die DSL.**
[3 Punkte]

5) Erzeugen Sie ein neues Dataframe indem Sie die originalen, eingelesenen Daten filtern, sodass nur mehr jene Baskets enthalten sind, welche aus „Chicago“, „Memphis“ od. „Baltimore“ stammen, mit „Amex“ bezahlt wurden und unter 200.00 Bestellwert aufweisen. Verwenden Sie dafür direkt **SQL und NICHT(!) die DSL.** *[3 Punkte]*

6) Berechnen Sie die **Anzahl an Baskets pro Bezahlungsmittel** („paymentType“) und sortieren Sie das Ergebnis aufsteigend nach Anzahl. Stellen Sie sicher, dass kein Ergebnis enthalten ist, bei dem das jeweilige Bezahlungsmittel weniger als 12 500 Baskets aufweist. Verwenden Sie dafür die **Spark DSL und KEINE(!) SQL Abfrage.** *[5 Punkte]*

7) Berechnen Sie **pro Kategorie der Baskets** (-> Feld „productCategory“) **den durchschnittlichen, minimalen sowie maximalen Bestellwert** (-> Feld „orderTotal“) **sortieren Sie das Ergebnis absteigend nach durchschnittlichem Bestellwert**. Sie können wahlweise zwischen SQL od. DSL wählen. *[6 Punkte]*

2 OpenFlights Data Analysis mit GraphFrames

1) Sie finden im package *edu.campus02.iwi.prfg* bereits die **Klasse FlightsAnalyzer.** Es ist bereits Code zum Einlesen für eine benötigte Input-Datei vorhanden. Laden Sie ebenso die zwei weiteren benötigten Dateien jeweils als Dataframe bzw. Dataset<Row>. *[2 Punkte]*

2) Erzeugen Sie danach ein GraphFrame auf Basis der Flughäfen (Knoten) sowie den Flugruten (Kanten) erstellen. *[1 Punkt]*

3) Finden Sie mittels GraphFrame sowie Spark SQL/DSL **alle Flughäfen in Großbritannien („`United Kingdom`“) oder der Australien („`Australia`“), die max. 100 ausgehende(!) Flugverbindungen** aufweisen. Geben Sie pro Ergebnis Datensatz Name,City sowie FlightCount der Flughäfen an. Das Ergebnis soll absteigend nach Anzahl an Flugverbindungen sortiert sein. *[5 Punkte]*

Hinweis: Sie müssen eine Join-Operation durchführen, damit Sie die benötigten Informationen zur Anreicherung für Filter nach Country sowie für die Ausgabe bekommen.

4) **Finden Sie alle verfügbaren direkten Flüge, welche von Norwegen nach Italien führen**. Geben Sie pro Ergebnisdatensatz die Start-Stadt, Ziel-Stadt sowie den Namen der Fluglinie an, die den Flug durchführt. *8 Punkte]*

Hinweis: Sie können diesen Task auf 2 verschiedene Arten lösen:

a) entweder Sie arbeiten direkt mit den entsprechenden Dataframes sowie JOINs
b) oder Sie verwenden für einen Teil ein GraphFrame mit passendem Motif-Finding