

# Beispiel 1: Kostenaufschläge Digital Marketing

## Aufgabenstellung:

Ein Medienunternehmen erbringt für seine Kunden Data Analyse Dienstleistungen und stellt die Kosten für die Digital Marketing Kampagnen in Dashboards dar, auf die der Kunde zugreifen kann. Den Kampagnenkosten sollen für diese Dienstleistung monatlich Kostenaufschläge hinzugerechnet werden, jeweils für Facebook und Instagram. Diese Aufschläge sind je nach Monatsplan in unterschiedlichen Höhen.

Der Kostenaufschlag soll einmalig jeden Monat auf den jeweils ersten Kampagneneintrag zu Facebook sowie zu Instagram aufgeschlagen werden. Der Kostenaufschlag soll automatisch erfolgen. Dazu wird das adverity transformation scripting sowie python expressions verwendet.

## Challenge/Problemstellung:

Die Kampagneneinträge zu Facebook und Instagram sind in der Datenbank jeden Monat unterschiedlich angeordnet. Daher muss zuerst herausgefunden werden, an welchen Positionen der Datenbank die ersten Einträge je Monat für Facebook sowie Instagram vorkommen. Im nächsten Schritt werden die dazugehörigen Kostenaufschläge hinzugerechnet.

## Vorbereitung:

Zuerst wird die Datenbank des data streams nach Datum und nach Publisher sortiert sowie eine zusätzliche Spalte gebildet, die das Datum des vorherigen Datensatzes beinhaltet.

Custom Script Configuration

sort

Enabled Add Comment ? Help X Remove

Key

- x String date\_start
- x String publisher\_platform
- +

Reverse

Subtable

Dann wird eine weitere Spalte gebildet, die den Publisher des vorherigen Datensatzes beinhaltet. Durch diese beiden Hilfsspalten wird die Suche nach den jeweils ersten Datensätze je Monat und Publisher wesentlich erleichtert.

addfieldusingcontextx

Enabled Add Comment ? Help X Remove

Field Name

prev\_platform

Python Expression

{prev['publisher\_platform']}

Subtable

Es wird eine neue Spalte für die Kosten gebildet, die die neuen Kosten inklusive Aufschlag beinhaltet.

addfieldx

Field Name: spend\_new

Python Expression: [{spend}]

Field Index: -1

Subtable:

Enabled Add Comment ? Help X Remove

Im nächsten Schritt wird die Position des ersten Datensatzes im Juni festgestellt und die Kostenaufschläge für Facebook sowie Instagram hinzugefügt:

convertx

and str({date\_start})[5:7] == '06'

Field: String spend\_new

Python Expression: float({spend\_new})+1000 if {date\_start}[-2:] == '02' and str({prev date})[-2:] != '02' and "facebook" in {publisher\_platform} and str({date\_start})[5:7] == '06' else {spend\_new}

Fail on Error: ☒

Subtable:

Enabled Remove Comment ? Help X Remove

#### Code für Kostenaufschlag von EUR 1.000 auf Facebook:

```
float({spend_new})+1000 if {date_start}[-2:] == '02' and str({prev date})[-2:] != '02' and "facebook" in {publisher_platform} and str({date_start})[5:7] == '06' else {spend_new}
```

#### Code für Kostenaufschlag von EUR 1.000 auf Instagram:

```
float({spend_new})+1000 if {date_start}[-2:] == '02' and str({prev_platform}) != 'instagram' and "instagram" in {publisher_platform} and str({date_start})[5:7] == '06' else {spend_new}
```

#### Ergebnis und Test der Umsetzung:

date_start	publisher_platform	spend	prev date	spend_new	prev_platform
2021-06-02	facebook	106,749		1106,749	
2021-06-02	facebook	3,9726	2021-06-02	3,9726	facebook
2021-06-02	facebook	13,0384	2021-06-02	13,0384	facebook
2021-06-02	facebook	0,03	2021-06-02	0,03	facebook
2021-06-02	facebook	0	2021-06-02	0	facebook
2021-06-02	instagram	59,446	2021-06-02	1059,446	facebook
2021-06-02	instagram	54,774	2021-06-02	54,774	instagram
2021-06-03	facebook	126,6433	2021-06-02	126,6433	instagram

Wie man hier sehen kann, wurde der erste Datensatz im Juni korrekt ermittelt und wie geplant die Aufschläge für Facebook sowie Instagram hinzugerechnet (grün markiert).

Im nächsten Schritt werden die ersten Datensätze für August und September ermittelt und die Kostenaufschläge für beide Monate jeweils für Facebook und Instagram hinzugefügt:

convertx

Enabled

Remove Comment

? Help

✖ Remove

▼

Speziell Facebook für August und September

Field

String

spend\_new

Python Expression

float({spend\_new})+3100 if {date\_start}[-2:] == '01' and (str({prev\_platform}) == 'audience\_network') and "facebook" in {publisher\_platform} an

Fail on Error

☒

Subtable

### Code für Kostenaufschlag von EUR 3.100 auf Facebook:

```
float({spend_new})+3100 if {date_start}[-2:] == '01' and (str({prev_platform}) == 'audience_network') and "facebook" in {publisher_platform} and str({date_start})[5:7] in ['08','09'] else {spend_new}
```

convertx

Enabled

Remove Comment

? Help

✖ Remove

▼

Speziell Instagram für August und September

Field

String

spend\_new

Python Expression

float({spend\_new})+3100 if {date\_start}[-2:] == '01' and str({prev\_platform}) == 'facebook' and "instagram" in {publisher\_platform} and str({da

Fail on Error

☒

Subtable

### Code für Kostenaufschlag von EUR 3.100 auf Instagram:

```
float({spend_new})+3100 if {date_start}[-2:] == '01' and str({prev_platform}) == 'facebook' and "instagram" in {publisher_platform} and str({date_start})[5:7] in ['08','09'] else {spend_new}
```

### Ergebnis und Test der Umsetzung:

date_start	publisher_platform	spend	prev date	spend_new	prev_platform
2021-08-01	audience_network	15,51	2021-08-01	15,51	audience_network
2021-08-01	audience_network	2,72	2021-08-01	2,72	audience_network
2021-08-01	facebook	59,27	2021-08-01	3159,27	audience_network
2021-08-01	facebook	0,02	2021-08-01	0,02	facebook
2021-08-01	facebook	3,87	2021-08-01	3,87	facebook
2021-08-01	instagram	1,51	2021-08-01	3101,51	facebook
2021-08-01	instagram	66,11	2021-08-01	66,11	instagram
2021-08-01	instagram	0	2021-08-01	0	instagram

Wie man hier sehen kann, wurden korrekt die ersten Datensätze im August für Facebook sowie Instagram identifiziert und wie geplant der Aufschlag von jeweils EUR 3.100 beiden Publisher hinzugerechnet (gelb markiert). Die Aufschläge für September wurden auch korrekt umgesetzt.