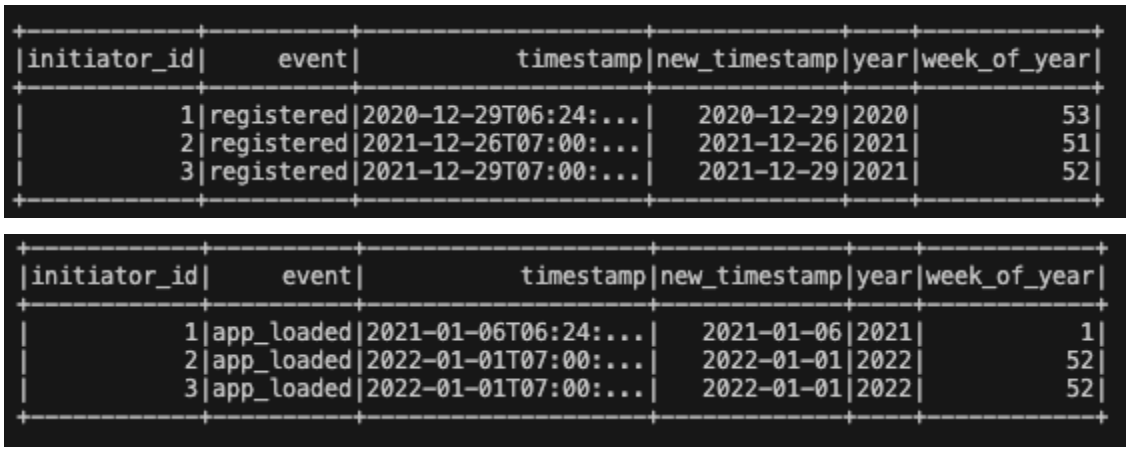
According to the pyspark documentation for weekofyear function, the function follows the ISO week numbering System.

ISO Week Numbering System
The ISO week numbering system follows these rules:

- A week starts on Monday.
- Week 1 is the first week of the year with at least four days in January.
- Years can have 52 or 53 weeks depending on how the weeks align with the new year.

Year End Edge Cases
As a result, there are edge cases that need to be accounted for registrations that occur at the end of the year i.e. at the last week of the year. When the registration occurs in the last week of one year, and the app is loaded in the first week of the next year.



initiator_id	event	timestamp	new_timestamp	year	week_of_year
1	registered	2020-12-29T06:24:...	2020-12-29	2020	53
2	registered	2021-12-26T07:00:...	2021-12-26	2021	51
3	registered	2021-12-29T07:00:...	2021-12-29	2021	52

initiator_id	event	timestamp	new_timestamp	year	week_of_year
1	app_loaded	2021-01-06T06:24:...	2021-01-06	2021	1
2	app_loaded	2022-01-01T07:00:...	2022-01-01	2022	52
3	app_loaded	2022-01-01T07:00:...	2022-01-01	2022	52

Subcase #1: (Initiator_1)

The app is loaded in week 1 of 2021, and registration was in week 53 of 2020, this example demonstrates a transition from the end of one year to the beginning of the next year.

Subcase #2: (Initiator_2)

The app is loaded in week 52 of 2021, while the registration was in week 51 of 2021, demonstrating the transition between weeks 51 and 52 of consecutive years despite the fact that date of app loaded is falling in year 2022.

Subcase #3: (initiator_3)

The app and registration both occur in week 52 of 2021 despite the date of app load falling in the year 2022, demonstrating that events in the same week but different years are not considered as per the requirement.

Other Edge Cases

App loaded more than once

In cases where app is loaded more than once, the first occurrence of app load event should be taken into account.