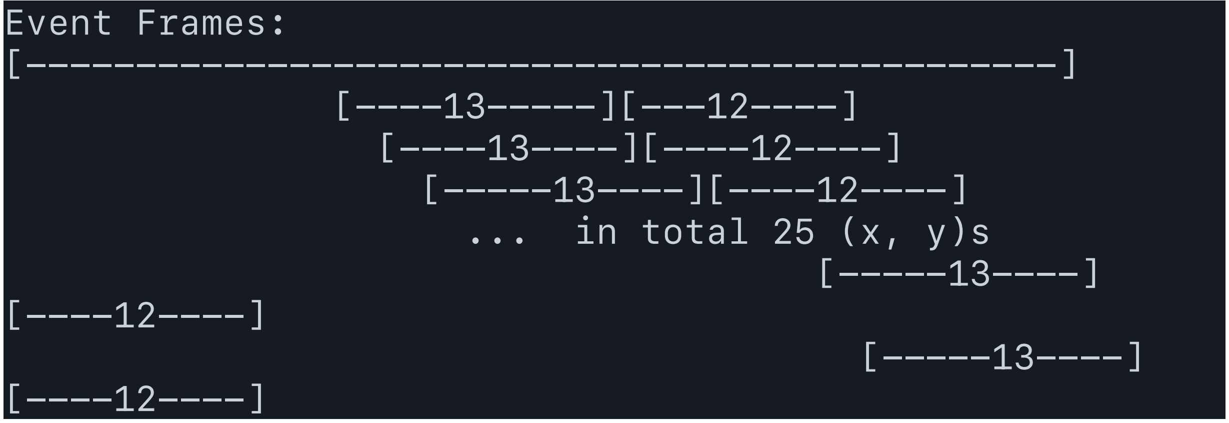
(Input parameters) -> filtering function() -> datapipeline() -> predicrt() -> visualize() ->. API_endpoint()

filtering function() -> Input: City, Date, Time
-> Output: Filename, fileindex, Eventid
-> Description: We take 4 parameters from user and search the that in our catalog files and if matched we will return filename in which the event is present else “no file found”

datapipeline() -> input: Filename,fileinded, event_id
->output: Test data IN : 13 images OUT: 12 images
-> Description: We take the filename that filtering function and lookup in our buckets the file, download that file and generate test data.
We are not using nowcast_test_generator.py as it has high scope we have created our own functions “split_data” to generate test data
We are dividing the 48 images in to 25 frames



```
Event Frames:
[-----------------------------------------------]
               [----13-----][---12----]
                 [----13----][----12----]
                   [-----13----][----12----]
                     ...  in total 25 (x, y)s
                                     [-----13----]
[----12----]
                                       [-----13----]
[----12----]
```

predict() -> input: test Data, Model
-> output: 12 Images
Description: Takes input as test data and uses model specified to predict 12 images

visualize() -> input 12 images

Api endpoints: nowcast_forecasting/forecast

nowcast_forecasting/backtest: 1) 12 Different images(path) 2) The array file h5 3) Analysis backtest image nowcast_forecasting/forecast 1) 112 Different images(path) 2) The array file h5