# FINAL CODE

| | |
|---|---|
| ***DATE*** | 6 NOVEMBER 2022 |
| ***TEAM ID*** | PNT2022TMID26602 |
| ***PROJECT NAME*** | Smart solutions for Railways |

## CODE:

```
# Import common libraries
import numpy as np import
pandas as pd import
matplotlib.pyplot as plt

# Import the PyGeohydro libaray
tools import pygeohydro as gh
from pygeohydro import SSFR, plot
```

```
# Use the smart solution for railways(SSFR)
ssfr = SSFR() # Specify date range of
interest dates = ("2020-01-01", "2020-12-
31")

# Filter stations to have only those with proper dates
stations = info_box[(info_box.begin_date <= dates[0]) &
(info_box.end_date >= dates[1])].site_no.tolist()

# Remove duplicates by converting to
a set stations = set(stations) # Specify
characteristics of interest

select_attributes = journey time ,train announcement ,
waiting arrangement ,security in the station, seat condition

# Initialize a storage matrix
nldi_data = np.zeros((len(flow_data.columns), len(select_attributes)))

# Loop through all gages, and request NLDI data near each gage
for i, st in enumerate(flow_data.columns):
```

```
    # Navigate up all flowlines from gage
    flowlines = NLDI().navigate_byid(fsource = 'nwissite',
                            fid = f'{st}',
                            navigation="upstreamTributaries"
                            , source = 'flowlines', distance =
                            10)

    # Get the nearest comid
    station_comid = flowlines.nhdplus_comid.to_list()[0]


    # Source NLDI local data
    nldi_data[i,:] = NLDI().getcharacteristic_byid(station_comid,
"local", char_ids = select_attributes)
```