How to add a new dataset to CFTree

1. ipynb file by loading the csv and set correct target columntree view for initial data
    a. after running the ipynb file, we get a pickle file dice_exp_{datasetName}.pkl: for example, dice_exp_LIDC.pkl
    b. copy this file into data folder of fastAPIbackend
2. add it DATASETS config in main.py:

```
DATASETS = {
    "income": {
        "df_cfs": "data/dice_exp_income.pkl",
        "columns_to_drop": ["income", "Original_Index"],
        "ordinal_attributes": {
            "education": ["School", "HS-grad", "Some-college", "Bachelors", "M
        },
        "target_column_name": "income",
    },
    "heart": {
        "df_cfs": "data/dice_exp_heart.pkl",
        "columns_to_drop": ["target", "Original_Index"],
        "ordinal_attributes": {
            "cp": ["Typical angina", "Atypical angina", "Non-anginal pain", "Asy
            "restecg": ["Normal", "ST-T wave abnormality", "Left ventricular hy
            "thal": ["Normal", "Fixed defect", "Reversible defect"],
            "slope": ["Upsloping", "Flat", "Downsloping"],
            "ca": ["0", "1", "2", "3"]
        },
        "target_column_name": "target",
    }
}
```

3. add a dataset target name we used (for example: diagnosis) to NavBar.js:

```
const datasetTargetNames = {
  income: "income",
  heart: "diagnosis",
};
```

4. for originalData in cfRows.js:
    a. {datasetKey}_x_test_first_20_with_target.json
4. There is no initial data for the group view. To ensure a good appearance on the first run, we can start the grouping on the frontend and then copy the backend-generated data for future initial use.

    in the group view, select one to generate the cfrows (copy backend's last_cfRowData to frontend's cfRowData.js) and treeresult (copy backend's last_tree_data.js to frontend's treeResult.js) for the initial group data