```
import pandas as pd
import numpy as np

df=pd.read_csv("train.csv")

df.head()
```


```
{"summary":"{\n  \"name\": \"df\",\n  \"rows\": 891,\n  \"fields\": [\
n    {\n      \"column\": \"PassengerId\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 257,\n        \"min\": 1,\n
\"max\": 891,\n        \"num_unique_values\": 891,\n
\"samples\": [\n          710,\n          440,\n          841\n
],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
}\n    },\n    {\n      \"column\": \"Survived\",\n
\"properties\": {\n        \"dtype\": \"number\",\n        \"std\":
0,\n        \"min\": 0,\n        \"max\": 1,\n
\"num_unique_values\": 2,\n        \"samples\": [\n          1,\n
0\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Pclass\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 0,\n        \"min\": 1,\n        \"max\": 3,\n
\"num_unique_values\": 3,\n        \"samples\": [\n          3,\n
1\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Name\",\n      \"properties\": {\n        \"dtype\": \"string\",\n
\"num_unique_values\": 891,\n        \"samples\": [\n
\"Moubarek, Master. Halim Gonios (\\\"William George\\\")\",\n
\"Kvillner, Mr. Johan Henrik Johannesson\"\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"Sex\",\n      \"properties\": {\n
\"dtype\": \"category\",\n        \"num_unique_values\": 2,\n
\"samples\": [\n          \"female\",\n          \"male\"\n        ],\
n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
}\n    },\n    {\n      \"column\": \"Age\",\n      \"properties\": {\
n        \"dtype\": \"number\",\n        \"std\": 14.526497332334042,\
n        \"min\": 0.42,\n        \"max\": 80.0,\n
\"num_unique_values\": 88,\n        \"samples\": [\n          0.75,\n
22.0\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"SibSp\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 1,\n        \"min\": 0,\n        \"max\": 8,\n
\"num_unique_values\": 7,\n        \"samples\": [\n          1,\n
0\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Parch\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 0,\n        \"min\": 0,\n        \"max\": 6,\n
\"num_unique_values\": 7,\n        \"samples\": [\n          0,\n
1\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Ticket\",\n      \"properties\": {\n        \"dtype\": \"string\",\n
```

```
\"num_unique_values\": 681,\n        \"samples\": [\n
\"11774\",\n          \"248740\"\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"Fare\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 49.6934285971809,\n
\"min\": 0.0,\n        \"max\": 512.3292,\n
\"num_unique_values\": 248,\n        \"samples\": [\n
11.2417,\n          51.8625\n        ],\n        \"semantic_type\":
\"\",\n        \"description\": \"\"\n      }\n    },\n    {\n
\"column\": \"Cabin\",\n      \"properties\": {\n        \"dtype\":
\"category\",\n        \"num_unique_values\": 147,\n
\"samples\": [\n          \"D45\",\n          \"B49\"\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"Embarked\",\n      \"properties\":
{\n        \"dtype\": \"category\",\n        \"num_unique_values\":
3,\n        \"samples\": [\n          \"S\",\n          \"C\"\n
],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
}\n    }\n  ]\n}","type":"dataframe","variable_name":"df"}
```


```
df.dtypes
```

```
PassengerId      int64
Survived         int64
Pclass           int64
Name            object
Sex             object
Age            float64
SibSp            int64
Parch            int64
Ticket          object
Fare           float64
Cabin           object
Embarked        object
dtype: object
```

```
categorical_columns=df.select_dtypes(include=['object','category']).co
lumns.tolist()
numerical_columns=df.select_dtypes(include=['int64','float64']).column
s.tolist()

print(f"categorical columns:\n (categorical_columns)")
print(f"numerical columns:\n (numerical_columns)")
```

```
categorical columns:
 (categorical_columns)
numerical columns:
 (numerical_columns)
```

```
columns_used
=['PassengerId','Survived','Pclass','Age','SibSp','Parch','Fare']
df1=df.loc[:,columns_used]

df1.head()
```


```
{"summary":"{\n  \"name\": \"df1\",\n  \"rows\": 891,\n  \"fields\":
[\n    {\n      \"column\": \"PassengerId\",\n      \"properties\": {\
n        \"dtype\": \"number\",\n        \"std\": 257,\n
\"min\": 1,\n        \"max\": 891,\n        \"num_unique_values\":
891,\n        \"samples\": [\n          710,\n          440,\n
841\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Survived\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 0,\n        \"min\": 0,\n
\"max\": 1,\n        \"num_unique_values\": 2,\n        \"samples\":
[\n          1,\n          0\n        ],\n        \"semantic_type\":
\"\",\n        \"description\": \"\"\n      }\n    },\n    {\n
\"column\": \"Pclass\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 0,\n        \"min\": 1,\n
\"max\": 3,\n        \"num_unique_values\": 3,\n        \"samples\":
[\n          3,\n          1\n        ],\n        \"semantic_type\":
\"\",\n        \"description\": \"\"\n      }\n    },\n    {\n
\"column\": \"Age\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 14.526497332334042,\n        \"min\":
0.42,\n        \"max\": 80.0,\n        \"num_unique_values\": 88,\n
\"samples\": [\n          0.75,\n          22.0\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"SibSp\",\n      \"properties\": {\
n        \"dtype\": \"number\",\n        \"std\": 1,\n        \"min\":
0,\n        \"max\": 8,\n        \"num_unique_values\": 7,\n
\"samples\": [\n          1,\n          0\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"Parch\",\n      \"properties\": {\
n        \"dtype\": \"number\",\n        \"std\": 0,\n        \"min\":
0,\n        \"max\": 6,\n        \"num_unique_values\": 7,\n
\"samples\": [\n          0,\n          1\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"Fare\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 49.6934285971809,\n
\"min\": 0.0,\n        \"max\": 512.3292,\n
\"num_unique_values\": 248,\n        \"samples\": [\n
11.2417,\n          51.8625\n        ],\n        \"semantic_type\":
\"\",\n        \"description\": \"\"\n      }\n    }\n  ]\
n}","type":"dataframe","variable_name":"df1"}
```


```
<google.colab._quickchart_helpers.SectionTitle at 0x7e4ecd48ada0>
```

```
from matplotlib import pyplot as plt
_df_0['index'].plot(kind='hist', bins=20, title='index')
plt.gca().spines[['top', 'right',]].set_visible(False)
```

```
from matplotlib import pyplot as plt
_df_1['PassengerId'].plot(kind='hist', bins=20, title='PassengerId')
plt.gca().spines[['top', 'right',]].set_visible(False)
```

```
from matplotlib import pyplot as plt
_df_2['Survived'].plot(kind='hist', bins=20, title='Survived')
plt.gca().spines[['top', 'right',]].set_visible(False)
```

```
from matplotlib import pyplot as plt
_df_3['Pclass'].plot(kind='hist', bins=20, title='Pclass')
plt.gca().spines[['top', 'right',]].set_visible(False)
```

```
<google.colab._quickchart_helpers.SectionTitle at 0x7e4ecd48be50>
```

```
from matplotlib import pyplot as plt
_df_4.plot(kind='scatter', x='index', y='PassengerId', s=32, alpha=.8)
plt.gca().spines[['top', 'right',]].set_visible(False)
```

```
from matplotlib import pyplot as plt
_df_5.plot(kind='scatter', x='PassengerId', y='Survived', s=32,
alpha=.8)
plt.gca().spines[['top', 'right',]].set_visible(False)
```

```
from matplotlib import pyplot as plt
_df_6.plot(kind='scatter', x='Survived', y='Pclass', s=32, alpha=.8)
plt.gca().spines[['top', 'right',]].set_visible(False)
```

```
from matplotlib import pyplot as plt
_df_7.plot(kind='scatter', x='Pclass', y='Age', s=32, alpha=.8)
plt.gca().spines[['top', 'right',]].set_visible(False)
```

```
<google.colab._quickchart_helpers.SectionTitle at 0x7e4eced313c0>
```

```
from matplotlib import pyplot as plt
import seaborn as sns
def _plot_series(series, series_name, series_index=0):
  palette = list(sns.palettes.mpl_palette('Dark2'))
  xs = series['index']
  ys = series['Survived']

  plt.plot(xs, ys, label=series_name, color=palette[series_index %
len(palette)])

fig, ax = plt.subplots(figsize=(10, 5.2), layout='constrained')
df_sorted = _df_8.sort_values('index', ascending=True)
_plot_series(df_sorted, '')
sns.despine(fig=fig, ax=ax)
```

```
plt.xlabel('index')
_ = plt.ylabel('Survived')
```

```
from matplotlib import pyplot as plt
import seaborn as sns
def _plot_series(series, series_name, series_index=0):
  palette = list(sns.palettes.mpl_palette('Dark2'))
  xs = series['index']
  ys = series['Pclass']

  plt.plot(xs, ys, label=series_name, color=palette[series_index %
len(palette)])

fig, ax = plt.subplots(figsize=(10, 5.2), layout='constrained')
df_sorted = _df_9.sort_values('index', ascending=True)
_plot_series(df_sorted, '')
sns.despine(fig=fig, ax=ax)
plt.xlabel('index')
_ = plt.ylabel('Pclass')
```

```
from matplotlib import pyplot as plt
import seaborn as sns
def _plot_series(series, series_name, series_index=0):
  palette = list(sns.palettes.mpl_palette('Dark2'))
  xs = series['index']
  ys = series['Age']

  plt.plot(xs, ys, label=series_name, color=palette[series_index %
len(palette)])

fig, ax = plt.subplots(figsize=(10, 5.2), layout='constrained')
df_sorted = _df_10.sort_values('index', ascending=True)
_plot_series(df_sorted, '')
sns.despine(fig=fig, ax=ax)
plt.xlabel('index')
_ = plt.ylabel('Age')
```

```
from matplotlib import pyplot as plt
import seaborn as sns
def _plot_series(series, series_name, series_index=0):
  palette = list(sns.palettes.mpl_palette('Dark2'))
  xs = series['index']
  ys = series['SibSp']

  plt.plot(xs, ys, label=series_name, color=palette[series_index %
len(palette)])

fig, ax = plt.subplots(figsize=(10, 5.2), layout='constrained')
df_sorted = _df_11.sort_values('index', ascending=True)
_plot_series(df_sorted, '')
```

```
sns.despine(fig=fig, ax=ax)
plt.xlabel('index')
_ = plt.ylabel('SibSp')
```

```
<google.colab._quickchart_helpers.SectionTitle at 0x7e4eced31e40>
```

```
from matplotlib import pyplot as plt
_df_12['index'].plot(kind='line', figsize=(8, 4), title='index')
plt.gca().spines[['top', 'right']].set_visible(False)
```

```
from matplotlib import pyplot as plt
_df_13['PassengerId'].plot(kind='line', figsize=(8, 4),
title='PassengerId')
plt.gca().spines[['top', 'right']].set_visible(False)
```

```
from matplotlib import pyplot as plt
_df_14['Survived'].plot(kind='line', figsize=(8, 4), title='Survived')
plt.gca().spines[['top', 'right']].set_visible(False)
```

```
from matplotlib import pyplot as plt
_df_15['Pclass'].plot(kind='line', figsize=(8, 4), title='Pclass')
plt.gca().spines[['top', 'right']].set_visible(False)
```

```
# prompt: Using dataframe df1: mean

# Calculate the mean of all numerical columns in the dataframe
df1.mean()
```

```
     PassengerId  Survived  Pclass   Age  SibSp  Parch  Fare
0              1       0.0     3.0  24.0    0.0    0.0  8.05
1              2       NaN     NaN   NaN    NaN    NaN   NaN
2              3       NaN     NaN   NaN    NaN    NaN   NaN
3              4       NaN     NaN   NaN    NaN    NaN   NaN
4              5       NaN     NaN   NaN    NaN    NaN   NaN
..           ...       ...     ...   ...    ...    ...   ...
886          887       NaN     NaN   NaN    NaN    NaN   NaN
887          888       NaN     NaN   NaN    NaN    NaN   NaN
888          889       NaN     NaN   NaN    NaN    NaN   NaN
889          890       NaN     NaN   NaN    NaN    NaN   NaN
890          891       NaN     NaN   NaN    NaN    NaN   NaN

[891 rows x 7 columns]
```

```
df1.std()
```

```
PassengerId    257.353842
Survived         0.486592
Pclass           0.836071
Age             14.526497
SibSp            1.102743
Parch            0.806057
```

```
Fare            49.693429
dtype: float64
```

```
modes = df1.mode()

# Print the modes
print(modes)
```

```
     PassengerId  Survived  Pclass   Age  SibSp  Parch  Fare
0              1       0.0     3.0  24.0    0.0    0.0  8.05
1              2       NaN     NaN   NaN    NaN    NaN   NaN
2              3       NaN     NaN   NaN    NaN    NaN   NaN
3              4       NaN     NaN   NaN    NaN    NaN   NaN
4              5       NaN     NaN   NaN    NaN    NaN   NaN
..           ...       ...     ...   ...    ...    ...   ...
886          887       NaN     NaN   NaN    NaN    NaN   NaN
887          888       NaN     NaN   NaN    NaN    NaN   NaN
888          889       NaN     NaN   NaN    NaN    NaN   NaN
889          890       NaN     NaN   NaN    NaN    NaN   NaN
890          891       NaN     NaN   NaN    NaN    NaN   NaN

[891 rows x 7 columns]
```

```
df1.median()
```

```
PassengerId    446.0000
Survived         0.0000
Pclass           3.0000
Age             28.0000
SibSp            0.0000
Parch            0.0000
Fare            14.4542
dtype: float64
```

```
df1.describe()
```


```
{"summary":"{\n  \"name\": \"df1\",\n  \"rows\": 8,\n  \"fields\": [\n{\n      \"column\": \"PassengerId\",\n      \"properties\": {\n\"dtype\": \"number\",\n        \"std\": 320.8159711429855,\n\"min\": 1.0,\n        \"max\": 891.0,\n        \"num_unique_values\": 6,\n        \"samples\": [\n          891.0,\n          446.0,\n668.5\n        ],\n        \"semantic_type\": \"\",\n\"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"Survived\",\n      \"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 314.8713661874558,\n        \"min\": 0.0,\n        \"max\": 891.0,\n        \"num_unique_values\": 5,\n\"samples\": [\n          0.3838383838383838,\n          1.0,\n0.4865924542648575\n        ],\n        \"semantic_type\": \"\",\n\"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"Pclass\",\n      \"properties\": {\n        \"dtype\": \"number\",\n\"std\": 314.2523437079694,\n        \"min\": 0.836071240977049,\n
```

```
\"max\": 891.0,\n        \"num_unique_values\": 6,\n
\"samples\": [\n          891.0,\n          2.308641975308642,\n
3.0\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Age\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 242.9056731818781,\n        \"min\": 0.42,\n        \"max\":
714.0,\n        \"num_unique_values\": 8,\n        \"samples\": [\n
29.69911764705882,\n          28.0,\n          714.0\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"SibSp\",\n      \"properties\": {\
n        \"dtype\": \"number\",\n        \"std\": 314.4908277465442,\n
\"min\": 0.0,\n        \"max\": 891.0,\n        \"num_unique_values\":
6,\n        \"samples\": [\n          891.0,\n
0.5230078563411896,\n          8.0\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"Parch\",\n      \"properties\": {\
n        \"dtype\": \"number\",\n        \"std\": 314.65971717879,\n
\"min\": 0.0,\n        \"max\": 891.0,\n        \"num_unique_values\":
5,\n        \"samples\": [\n          0.38159371492704824,\n
6.0,\n          0.8060572211299483\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"Fare\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 330.6256632228578,\n
\"min\": 0.0,\n        \"max\": 891.0,\n        \"num_unique_values\":
8,\n        \"samples\": [\n          32.204207968574636,\n
14.4542,\n          891.0\n        ],\n        \"semantic_type\":
\"\",\n        \"description\": \"\"\n      }\n    }\n  ]\
n}","type":"dataframe"}
```


```
df.head(2)
```


```
{"summary":"{\n  \"name\": \"df\",\n  \"rows\": 891,\n  \"fields\": [\
n    {\n      \"column\": \"PassengerId\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 257,\n        \"min\": 1,\n
\"max\": 891,\n        \"num_unique_values\": 891,\n
\"samples\": [\n          710,\n          440,\n          841\n
],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
}\n    },\n    {\n      \"column\": \"Survived\",\n
\"properties\": {\n        \"dtype\": \"number\",\n        \"std\":
0,\n        \"min\": 0,\n        \"max\": 1,\n
\"num_unique_values\": 2,\n        \"samples\": [\n          1,\n
0\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Pclass\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 0,\n        \"min\": 1,\n        \"max\": 3,\n
\"num_unique_values\": 3,\n        \"samples\": [\n          3,\n
1\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Name\",\n      \"properties\": {\n        \"dtype\": \"string\",\n
\"num_unique_values\": 891,\n        \"samples\": [\n
```

\"Moubarek, Master. Halim Gonios (\\\"William George\\\")\",\n\"Kvillner, Mr. Johan Henrik Johannesson\"\n   ],\n\"semantic_type\": \"\",\n   \"description\": \"\"\n   }\n   },\n   {\n   \"column\": \"Sex\",\n   \"properties\": {\n\"dtype\": \"category\",\n   \"num_unique_values\": 2,\n\"samples\": [\n   \"female\",\n   \"male\"\n   ],\n   \"semantic_type\": \"\",\n   \"description\": \"\"\n}\n   },\n   {\n   \"column\": \"Age\",\n   \"properties\": {\n   \"dtype\": \"number\",\n   \"std\": 14.526497332334042,\n   \"min\": 0.42,\n   \"max\": 80.0,\n\"num_unique_values\": 88,\n   \"samples\": [\n   0.75,\n22.0\n   ],\n   \"semantic_type\": \"\",\n\"description\": \"\"\n   }\n   },\n   {\n   \"column\": \"SibSp\",\n   \"properties\": {\n   \"dtype\": \"number\",\n\"std\": 1,\n   \"min\": 0,\n   \"max\": 8,\n\"num_unique_values\": 7,\n   \"samples\": [\n   1,\n0\n   ],\n   \"semantic_type\": \"\",\n\"description\": \"\"\n   }\n   },\n   {\n   \"column\": \"Parch\",\n   \"properties\": {\n   \"dtype\": \"number\",\n\"std\": 0,\n   \"min\": 0,\n   \"max\": 6,\n\"num_unique_values\": 7,\n   \"samples\": [\n   0,\n1\n   ],\n   \"semantic_type\": \"\",\n\"description\": \"\"\n   }\n   },\n   {\n   \"column\": \"Ticket\",\n   \"properties\": {\n   \"dtype\": \"string\",\n\"num_unique_values\": 681,\n   \"samples\": [\n\"11774\",\n   \"248740\"\n   ],\n\"semantic_type\": \"\",\n   \"description\": \"\"\n   }\n   },\n   {\n   \"column\": \"Fare\",\n   \"properties\": {\n\"dtype\": \"number\",\n   \"std\": 49.6934285971809,\n\"min\": 0.0,\n   \"max\": 512.3292,\n\"num_unique_values\": 248,\n   \"samples\": [\n11.2417,\n   51.8625\n   ],\n   \"semantic_type\": \"\",\n   \"description\": \"\"\n   }\n   },\n   {\n\"column\": \"Cabin\",\n   \"properties\": {\n   \"dtype\": \"category\",\n   \"num_unique_values\": 147,\n\"samples\": [\n   \"D45\",\n   \"B49\"\n   ],\n\"semantic_type\": \"\",\n   \"description\": \"\"\n   }\n   },\n   {\n   \"column\": \"Embarked\",\n   \"properties\": {\n   \"dtype\": \"category\",\n   \"num_unique_values\": 3,\n   \"samples\": [\n   \"S\",\n   \"C\"\n],\n   \"semantic_type\": \"\",\n   \"description\": \"\"\n}\n   }\n]\n}","type":"dataframe","variable_name":"df"}

```
df['Sex'].unique()
```

```
array(['male', 'female'], dtype=object)
```

```
df[['Sex','Age']].groupby('Sex').mean()
```

{"summary":"{\n \"name\": \"df[['Sex','Age']]\",\n \"rows\": 2,\n
\"fields\": [\n   {\n   \"column\": \"Sex\",\n
\"properties\": {\n   \"dtype\": \"string\",\n
\"num_unique_values\": 2,\n   \"samples\": [\n
\"male\",\n   \"female\"\n   ],\n
\"semantic_type\": \"\",\n   \"description\": \"\"\n   }\
n   },\n   {\n   \"column\": \"Age\",\n   \"properties\": {\n
\"dtype\": \"number\",\n   \"std\": 1.9876317510589439,\n
\"min\": 27.915708812260537,\n   \"max\": 30.72664459161148,\n
\"num_unique_values\": 2,\n   \"samples\": [\n
30.72664459161148,\n   27.915708812260537\n   ],\n
\"semantic_type\": \"\",\n   \"description\": \"\"\n   }\
n   }\n]\n}","type":"dataframe"}

```
df['Survived'].unique()
```

```
array([0, 1])
```

```
df[['Survived','Age']].groupby('Survived').mean()
```

{"summary":"{\n \"name\": \"df[['Survived','Age']]\",\n \"rows\":
2,\n \"fields\": [\n   {\n   \"column\": \"Survived\",\n
\"properties\": {\n   \"dtype\": \"number\",\n   \"std\":
0,\n   \"min\": 0,\n   \"max\": 1,\n
\"num_unique_values\": 2,\n   \"samples\": [\n   1,\n
0\n   ],\n   \"semantic_type\": \"\",\n
\"description\": \"\"\n   }\n   },\n   {\n   \"column\":
\"Age\",\n   \"properties\": {\n   \"dtype\": \"number\",\n
\"std\": 1.6139638671549115,\n   \"min\": 28.343689655172415,\n
\"max\": 30.62617924528302,\n   \"num_unique_values\": 2,\n
\"samples\": [\n   28.343689655172415,\n
30.62617924528302\n   ],\n   \"semantic_type\": \"\",\n
\"description\": \"\"\n   }\n   }\n]\n}","type":"dataframe"}