

Titanic

NO.

Library

```
{ import numpy as np
  import pandas as pd
  import seaborn as sns
  sns.set(style='darkgrid')
  style =
```

```
{ from sklearn.ensemble import RandomForestClassifier
  from sklearn.preprocessing import OneHotEncoder, LabelEncoder, StandardScaler
  from sklearn.metrics import roc_curve, auc
  from sklearn.model_selection import StratifiedKFold
```

```
import string
```

```
import warnings
```

```
warnings.filterwarnings('ignore')
```

```
SEED = 42
```

```
def concat_df(train_data, test_data):
```

```
    return pd.concat([train_data, test_data], sort=True).reset_index(
        drop=True)
```

```
def divide_df(all_data)
```

```
    return all_data.iloc[:890], all_data.iloc[890:].drop(['Survived'],
        axis=1)
```


1.2.1 Age.

```
df_all_corr = df_all.corr().abs().unstack().sort_values(
    kind='quicksort', ascending=False).reset_index()
```

```
df_all_corr.rename(columns={'axel_0': 'Feature 1', ... 3, inplace=True)
```

```
df_all_corr[df_all_corr['Feature 1'] == 'Age']
```

index 별

median

찾기

```
age_by_pclass_sex = df_all.groupby(['Sex', 'pclass']).median()['Age']
```

```
df_all['Age'] = df_all.groupby(['Sex', 'pclass'])['Age'].apply(lambda x:
    x.fillna(x.median()))
```

1.2.2 Embarked.

```
df_all[df_all['Embarked'].isnull()]
```

```
df_all['Embarked'] = df_all['Embarked'].fillna('S')
```

1.2.3 Fare

['Fare']

```
med_fare = df_all.groupby(['pclass', 'parch', 'sibsp'])['Fare'].median()[3][0][0]
```

['Fare']

1.2.4 Cabin.