

```
In [16]: import pandas as pk
```

```
In [49]: df = pk.read_csv("C:/Users/aziz/Desktop/Downloads/Machine Learning/salaries.csv")
df.head()
```

Out[49]:

	company	job	degree	salary_more_than_100k
0	google	sales executive	bachelors	0
1	google	sales executive	masters	0
2	google	business manager	bachelors	1
3	google	business manager	masters	1
4	google	computer programmer	bachelors	0

```
In [50]: inputs = df.drop('salary_more_than_100k',axis='columns')
```

```
In [19]: target =df['salary_more_than_100k']
```

```
In [20]: inputs
```

Out[20]:

	company	job	degree
0	google	sales executive	bachelors
1	google	sales executive	masters
2	google	business manager	bachelors
3	google	business manager	masters
4	google	computer programmer	bachelors
5	google	computer programmer	masters
6	abc pharma	sales executive	masters
7	abc pharma	computer programmer	bachelors
8	abc pharma	business manager	bachelors
9	abc pharma	business manager	masters
10	facebook	sales executive	bachelors
11	facebook	sales executive	masters
12	facebook	business manager	bachelors
13	facebook	business manager	masters
14	facebook	computer programmer	bachelors
15	facebook	computer programmer	masters

```
In [21]: target
```

```
Out[21]: 0      0
          1      0
          2      1
          3      1
          4      0
          5      1
          6      0
          7      0
          8      0
          9      1
         10      1
         11      1
         12      1
         13      1
         14      1
         15      1
          Name: salary_more_than_100k, dtype: int64
```

```
In [22]: from sklearn.preprocessing import LabelEncoder
```

```
In [34]: le_company = LabelEncoder()
          le_job = LabelEncoder()
          le_degree = LabelEncoder()
```

```
In [35]: inputs['company_n'] = le_company.fit_transform(inputs['company'])
          inputs['job_n'] = le_job.fit_transform(inputs['job'])
          inputs['degree_n'] = le_degree.fit_transform(inputs['degree'])
```

```
In [36]: inputs
```

```
Out[36]:
```

	company	job	degree	company_n	job_n	degree_n
0	google	sales executive	bachelors	2	2	0
1	google	sales executive	masters	2	2	1
2	google	business manager	bachelors	2	0	0
3	google	business manager	masters	2	0	1
4	google	computer programmer	bachelors	2	1	0
5	google	computer programmer	masters	2	1	1
6	abc pharma	sales executive	masters	0	2	1
7	abc pharma	computer programmer	bachelors	0	1	0
8	abc pharma	business manager	bachelors	0	0	0
9	abc pharma	business manager	masters	0	0	1
10	facebook	sales executive	bachelors	1	2	0
11	facebook	sales executive	masters	1	2	1
12	facebook	business manager	bachelors	1	0	0
13	facebook	business manager	masters	1	0	1
14	facebook	computer programmer	bachelors	1	1	0
15	facebook	computer programmer	masters	1	1	1

```
In [38]: inputs_n = inputs.drop(['company', 'job', 'degree'], axis='columns')
```

```
In [39]: inputs_n
```

```
Out[39]:
```

	company_n	job_n	degree_n
0	2	2	0
1	2	2	1
2	2	0	0
3	2	0	1
4	2	1	0
5	2	1	1
6	0	2	1
7	0	1	0
8	0	0	0
9	0	0	1
10	1	2	0
11	1	2	1
12	1	0	0
13	1	0	1
14	1	1	0
15	1	1	1

```
In [40]: target
```

```
Out[40]: 0      0
1      0
2      1
3      1
4      0
5      1
6      0
7      0
8      0
9      1
10     1
11     1
12     1
13     1
14     1
15     1
Name: salary_more_than_100k, dtype: int64
```

```
In [41]: from sklearn import tree
```

```
In [42]: model = tree.DecisionTreeClassifier()
```

```
In [43]: model.fit(inputs_n,target)
```

```
Out[43]: DecisionTreeClassifier(class_weight=None, criterion='gini', max_depth=None,  
                                max_features=None, max_leaf_nodes=None,  
                                min_impurity_decrease=0.0, min_impurity_split=None,  
                                min_samples_leaf=1, min_samples_split=2,  
                                min_weight_fraction_leaf=0.0, presort=False, random_state=None,  
                                splitter='best')
```

```
In [44]: model.score(inputs_n,target)
```

```
Out[44]: 1.0
```

```
In [45]: model.predict([[2,2,1]])
```

```
Out[45]: array([0], dtype=int64)
```

```
In [47]: model.predict([[2,1,1]])
```

```
Out[47]: array([1], dtype=int64)
```

```
In [ ]:
```