In [34]:
```
import pandas as pd
import numpy as np
from sklearn.preprocessing import LabelEncoder
```

In [35]:
```
df= pd.read_csv('adult.csv')
```

In [36]:
```
df
```

Out[36]:

	Age	Workclass	Final Weight	Education	EducationNum	Marital Status	Occupation	Relationship
0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family
1	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband
2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family
3	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband
4	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife
...	...	...	...	...	...	...	...	...
32556	27	Private	257302	Assoc-acdm	12	Married-civ-spouse	Tech-support	Wife
32557	40	Private	154374	HS-grad	9	Married-civ-spouse	Machine-op-inspct	Husband
32558	58	Private	151910	HS-grad	9	Widowed	Adm-clerical	Unmarried
32559	22	Private	201490	HS-grad	9	Never-married	Adm-clerical	Own-child
32560	52	Self-emp-inc	287927	HS-grad	9	Married-civ-spouse	Exec-managerial	Wife

32561 rows × 15 columns

In [37]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 32561 entries, 0 to 32560
Data columns (total 15 columns):
 #   Column          Non-Null Count  Dtype 
---  ------          --------------  ----- 
 0   Age             32561 non-null  int64 
 1   Workclass       32561 non-null  object
 2   Final Weight    32561 non-null  int64 
 3   Education       32561 non-null  object
 4   EducationNum    32561 non-null  int64 
 5   Marital Status  32561 non-null  object
 6   Occupation      32561 non-null  object
 7   Relationship    32561 non-null  object
 8   Race            32561 non-null  object
 9   Gender          32561 non-null  object
 10  Capital Gain    32561 non-null  int64 
 11  capital loss    32561 non-null  int64 
 12  Hours per Week  32561 non-null  int64 
 13  Native Country  32561 non-null  object
 14  Income          32561 non-null  object
dtypes: int64(6), object(9)
memory usage: 3.7+ MB
```

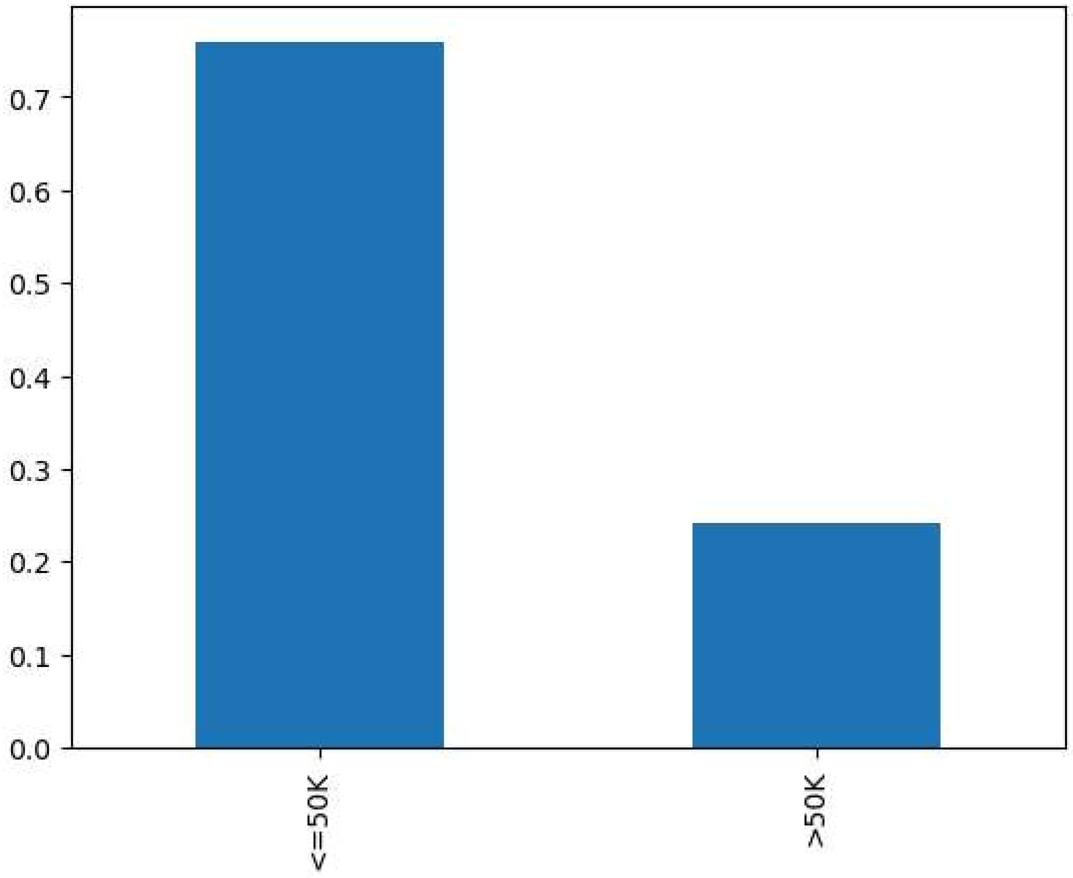
In [38]:
```
df.describe()
```

Out[38]:

	Age	Final Weight	EducationNum	Capital Gain	capital loss	Hours per Week
count	32561.000000	3.256100e+04	32561.000000	32561.000000	32561.000000	32561.000000
mean	38.581647	1.897784e+05	10.080679	1077.648844	87.303830	40.437456
std	13.640433	1.055500e+05	2.572720	7385.292085	402.960219	12.347429
min	17.000000	1.228500e+04	1.000000	0.000000	0.000000	1.000000
25%	28.000000	1.178270e+05	9.000000	0.000000	0.000000	40.000000
50%	37.000000	1.783560e+05	10.000000	0.000000	0.000000	40.000000
75%	48.000000	2.370510e+05	12.000000	0.000000	0.000000	45.000000
max	90.000000	1.484705e+06	16.000000	99999.000000	4356.000000	99.000000

In [39]:
```
df.Income.value_counts(normalize=True).plot(kind='bar')
```

Out[39]:
```
<Axes: >
```



In [40]:
```
df.describe()
```

Out[40]:

	Age	Final Weight	EducationNum	Capital Gain	capital loss	Hours per Week
count	32561.000000	3.256100e+04	32561.000000	32561.000000	32561.000000	32561.000000
mean	38.581647	1.897784e+05	10.080679	1077.648844	87.303830	40.437456
std	13.640433	1.055500e+05	2.572720	7385.292085	402.960219	12.347429
min	17.000000	1.228500e+04	1.000000	0.000000	0.000000	1.000000
25%	28.000000	1.178270e+05	9.000000	0.000000	0.000000	40.000000
50%	37.000000	1.783560e+05	10.000000	0.000000	0.000000	40.000000
75%	48.000000	2.370510e+05	12.000000	0.000000	0.000000	45.000000
max	90.000000	1.484705e+06	16.000000	99999.000000	4356.000000	99.000000

In [41]:
```
bins = [0, 12, 19, 35, 60, 100]
```

In [42]:
```
labels = ['Child', 'Teen', 'Adult', 'Middle-Aged', 'Senior']
```

In [43]:

```
df['Age_Group'] = pd.cut(df['Age'], bins=bins, labels=labels, right=False)
```

In [44]:

```
df
```

Out[44]:

	Age	Workclass	Final Weight	Education	EducationNum	Marital Status	Occupation	Relationship
0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family
1	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband
2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family
3	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband
4	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife
...	...	...	...	...	...	...	...	...
32556	27	Private	257302	Assoc-acdm	12	Married-civ-spouse	Tech-support	Wife
32557	40	Private	154374	HS-grad	9	Married-civ-spouse	Machine-op-inspct	Husband
32558	58	Private	151910	HS-grad	9	Widowed	Adm-clerical	Unmarried
32559	22	Private	201490	HS-grad	9	Never-married	Adm-clerical	Own-child
32560	52	Self-emp-inc	287927	HS-grad	9	Married-civ-spouse	Exec-managerial	Wife

32561 rows × 16 columns

In [45]:

```
grouped_data = df.groupby(['Income', 'Age_Group'])['Age'].mean()
```

In [46]:
```
grouped_data
```

Out[46]:
```
Income  Age_Group
<=50K   Child                NaN
        Teen           17.582011
        Adult          26.287534
        Middle-Aged    44.687518
        Senior         66.759538
>50K    Child                NaN
        Teen                 NaN
        Adult          30.370296
        Middle-Aged    45.470619
        Senior         65.251534
Name: Age, dtype: float64
```

In [47]:
```
grouped_data = df.groupby(['Income', 'Age_Group'])['Age'].describe()
```

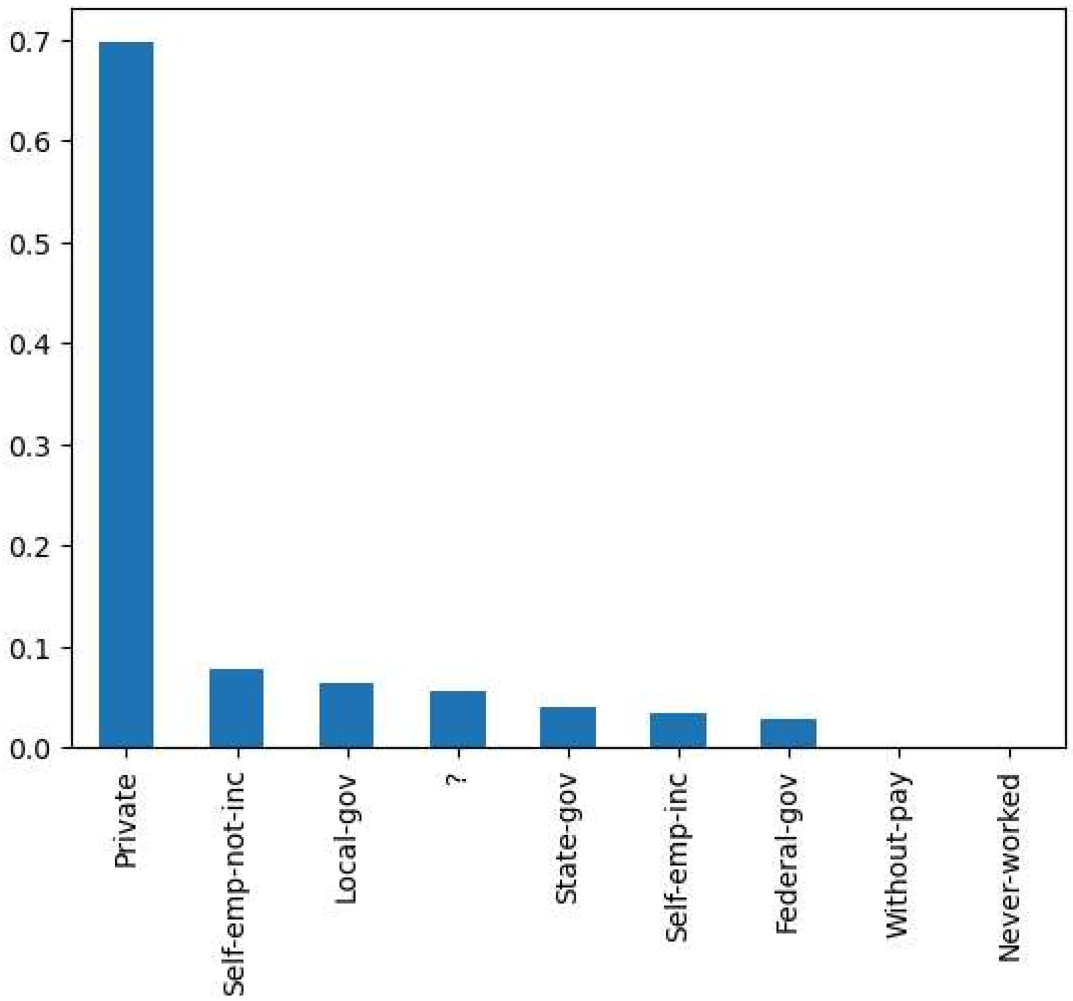
In [48]:
```
grouped_data
```

Out[48]:

		count	mean	std	min	25%	50%	75%	max
Income	**Age_Group**								
<=50K	**Teen**	945.0	17.582011	0.493490	17.0	17.0	18.0	18.0	18.0
	Adult	11616.0	26.287534	4.533604	19.0	22.0	26.0	30.0	34.0
	Middle-Aged	10167.0	44.687518	6.910392	35.0	39.0	44.0	50.0	59.0
	Senior	1992.0	66.759538	6.386663	60.0	62.0	65.0	70.0	90.0
>50K	**Adult**	1488.0	30.370296	2.960735	19.0	28.0	31.0	33.0	34.0
	Middle-Aged	5701.0	45.470619	6.637167	35.0	40.0	45.0	51.0	59.0
	Senior	652.0	65.251534	5.630555	60.0	61.0	64.0	67.0	90.0

In [49]:
```
df.Workclass.value_counts(normalize=True).plot(kind='bar')p
```

Out[49]:
```
<Axes: >
```



In [50]:
```
le = LabelEncoder()
# Apply label encoding to 'Income'
df['Income_encoded'] = le.fit_transform(df['Income'])
```

In [51]:

```
df
```

Out[51]:

	Age	Workclass	Final Weight	Education	EducationNum	Marital Status	Occupation	Relationship
0	39	State-gov	77516	Bachelors	13	Never-married	Adm-clerical	Not-in-family
1	50	Self-emp-not-inc	83311	Bachelors	13	Married-civ-spouse	Exec-managerial	Husband
2	38	Private	215646	HS-grad	9	Divorced	Handlers-cleaners	Not-in-family
3	53	Private	234721	11th	7	Married-civ-spouse	Handlers-cleaners	Husband
4	28	Private	338409	Bachelors	13	Married-civ-spouse	Prof-specialty	Wife
...	...	...	...	...	...	...	...	...
32556	27	Private	257302	Assoc-acdm	12	Married-civ-spouse	Tech-support	Wife
32557	40	Private	154374	HS-grad	9	Married-civ-spouse	Machine-op-inspct	Husband
32558	58	Private	151910	HS-grad	9	Widowed	Adm-clerical	Unmarried
32559	22	Private	201490	HS-grad	9	Never-married	Adm-clerical	Own-child
32560	52	Self-emp-inc	287927	HS-grad	9	Married-civ-spouse	Exec-managerial	Wife

32561 rows × 17 columns

In [ ]: