

```
In [17]: import pandas as pd  
import numpy as np
```

```
df = pd.read_csv("adult.csv")  
df
```

Out[17]:

	age	workclass	fnlwgt	education	education.num	marital.status	occupation
0	90	?	77053	HS-grad	9	Widowed	?
1	82	Private	132870	HS-grad	9	Widowed	Exec-managerial
2	66	?	186061	Some-college	10	Widowed	?
3	54	Private	140359	7th-8th	4	Divorced	Machine-op-inspt
4	41	Private	264663	Some-college	10	Separated	Prof-specialty
...
32556	22	Private	310152	Some-college	10	Never-married	Protective-serv
32557	27	Private	257302	Assoc-acdm	12	Married-civ-spouse	Tech-support
32558	40	Private	154374	HS-grad	9	Married-civ-spouse	Machine-op-inspt
32559	58	Private	151910	HS-grad	9	Widowed	Adm-clerical
32560	22	Private	201490	HS-grad	9	Never-married	Adm-clerical

32561 rows × 15 columns

In [19]:

```
df = df[["education", "age"]]  
age_list_by_education = df.groupby("education")["age"].apply(list)  
print("List of Age values grouped by Education:\n")  
age_list_by_education
```

List of Age values grouped by Education:

Out[19]:

```
education  
10th      [38, 37, 38, 59, 28, 47, 39, 35, 27, 51, 58, 5...  
11th      [45, 29, 50, 64, 19, 35, 43, 32, 48, 31, 27, 4...  
12th      [33, 39, 25, 17, 56, 19, 20, 43, 79, 39, 19, 6...  
1st-4th    [33, 58, 73, 57, 53, 61, 48, 32, 67, 68, 64, 5...  
5th-6th    [49, 46, 61, 69, 47, 43, 46, 56, 61, 26, 41, 3...  
7th-8th    [54, 42, 28, 41, 29, 56, 50, 24, 48, 40, 48, 6...  
9th       [49, 45, 24, 47, 17, 19, 60, 39, 43, 55, 45, 6...  
Assoc-acdm [22, 43, 40, 46, 20, 45, 34, 32, 28, 30, 25, 2...  
Assoc-voc  [21, 51, 33, 60, 41, 39, 26, 43, 41, 68, 29, 2...  
Bachelors [52, 34, 37, 51, 44, 48, 32, 38, 51, 39, 49, 5...  
Doctorate [74, 45, 51, 63, 37, 59, 46, 72, 72, 68, 66, 4...  
HS-grad    [90, 82, 34, 68, 61, 61, 60, 53, 71, 71, 73, 5...  
Masters   [32, 57, 54, 43, 43, 39, 37, 39, 46, 40, 31, 4...  
Preschool [26, 40, 27, 32, 53, 51, 71, 31, 33, 39, 52, 5...  
Prof-school [38, 46, 52, 51, 40, 41, 55, 42, 41, 60, 39, 4...  
Some-college [66, 41, 41, 51, 23, 51, 68, 67, 42, 53, 40, 4...  
Name: age, dtype: object
```

In [21]:

```
summary_statistics = df.groupby("education")["age"].agg(  
    Mean="mean",  
    Median="median",
```

```
    Minimum="min",
    Maximum="max",
    Standard_Deviation="std"
)
```

```
print("\nSummary Statistics of Age grouped by Education:\n")
summary_statistics
```

Summary Statistics of Age grouped by Education:

Out[21]:

education	Mean	Median	Minimum	Maximum	Standard_Deviation
10th	37.429796	34.0	17	90	16.720713
11th	32.355745	28.0	17	90	15.545485
12th	32.000000	28.0	17	79	14.334625
1st-4th	46.142857	46.0	19	90	15.615625
5th-6th	42.885886	42.0	17	84	15.557285
7th-8th	48.445820	50.0	17	90	16.092350
9th	41.060311	39.0	17	90	15.946862
Assoc-acdm	37.381443	36.0	19	90	11.095177
Assoc-voc	38.553546	37.0	19	84	11.631300
Bachelors	38.904949	37.0	19	90	11.912210
Doctorate	47.702179	47.0	24	80	11.784716
HS-grad	38.974479	37.0	17	90	13.541524
Masters	44.049913	43.0	18	90	11.068935
Preschool	42.764706	41.0	19	75	15.126914
Prof-school	44.746528	43.0	25	90	11.962477
Some-college	35.756275	34.0	17	90	13.474051

In [23]:

```
df = pd.read_csv("Iris.csv")
df
```

Out[23]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa
...
145	146	6.7	3.0	5.2	2.3	Iris-virginica
146	147	6.3	2.5	5.0	1.9	Iris-virginica
147	148	6.5	3.0	5.2	2.0	Iris-virginica
148	149	6.2	3.4	5.4	2.3	Iris-virginica
149	150	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 6 columns

In [27]:

```
df = df.drop(columns=["Id"], errors="ignore")

species_names = ["Iris-setosa", "Iris-versicolor", "Iris-virginica"]

for species in species_names:
    print(f"\nStatistical Details for {species}\n")

    species_data = df[df["Species"] == species].drop(columns=["Species"])

    # Mean
    print("Mean:")
    print(species_data.mean())

    # Standard Deviation
    print("\nStandard Deviation:")
    print(species_data.std())

    # Percentiles
    print("\nPercentiles (25%, 50%, 75%):")
    print(np.percentile(species_data, [25, 50, 75], axis=0))
```

Statistical Details for Iris-setosa

Mean:

```
SepalLengthCm    5.006
SepalWidthCm     3.418
PetalLengthCm    1.464
PetalWidthCm     0.244
dtype: float64
```

Standard Deviation:

```
SepalLengthCm    0.352490
SepalWidthCm     0.381024
PetalLengthCm    0.173511
PetalWidthCm     0.107210
dtype: float64
```

Percentiles (25%, 50%, 75%):

```
[[4.8   3.125 1.4   0.2   ]
 [5.    3.4    1.5   0.2   ]
 [5.2   3.675 1.575 0.3   ]]
```

Statistical Details for Iris-versicolor

Mean:

```
SepalLengthCm    5.936
SepalWidthCm     2.770
PetalLengthCm    4.260
PetalWidthCm     1.326
dtype: float64
```

Standard Deviation:

```
SepalLengthCm    0.516171
SepalWidthCm     0.313798
PetalLengthCm    0.469911
PetalWidthCm     0.197753
dtype: float64
```

Percentiles (25%, 50%, 75%):

```
[[5.6   2.525 4.     1.2   ]
 [5.9   2.8    4.35  1.3   ]
 [6.3   3.     4.6   1.5   ]]
```

Statistical Details for Iris-virginica

Mean:

```
SepalLengthCm    6.588
SepalWidthCm     2.974
PetalLengthCm    5.552
PetalWidthCm     2.026
dtype: float64
```

Standard Deviation:

```
SepalLengthCm    0.635880
SepalWidthCm     0.322497
PetalLengthCm    0.551895
PetalWidthCm     0.274650
dtype: float64
```

Percentiles (25%, 50%, 75%):

```
[[6.225 2.8   5.1   1.8   ]
 [6.5   3.     5.55  2.     ]
 [6.9   3.175 5.875 2.3   ]]
```

In []: