Introduction

The following presentation focuses on the reading habits of US citizens. The Pew Research Centre gathered the data we'll analyse in 2020 and considers information obtained from over 2000 samples. It is a dataset that was cited in several articles, and it is thus of considerable relevance. However, while it might appear valid on paper, we shall also make to sure to check its accuracy.

The report will focus particularly on:

- **Method of acquisition and variables**.
- **Graphical Analysis**
- **Insights and Critique**

Method of acquisition and variables

The data was gathered through a survey with multiple choices, and thus the variables we will deal with are the questions that were asked to the participants. While it was not disclosed whether the survey was in person or online, the nature of the questions, the number of samples gathered and the time period all suggest the survey was perfomed online.

The variables are thus:

1. Age
2. Sex
3. Race
4. Marital status?
5. Education
6. Employement
7. Incomes
8. How many books did you read during last 12 months?
9. Read any printed books during last 12 months?
10. Read any audiobooks during last 12 months?
11. Read any e-books during last 12 months?
12. Last book you read, you... (Note: the question is whether the last book read was bought or borrowed)
13. Do you happen to read any daily news or newspapers?
14. Do you happen to read any magazines or journals?

Note: the participants were allowed not to answer some of the questions(mostly those linked to identity), thus there will be a certain amount of null values.

```
# library for data

import numpy as np
import pandas as pd

# Library for kaggle
import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))


# library for visualisation
import matplotlib.pyplot as plt
import seaborn as sns

sns.set_style("whitegrid")
```

```
/kaggle/input/reading-habit-dataset/BigML_Dataset_5f50a62795a9306aa200003e.csv
```

```
# import data
df = pd.read_csv("/kaggle/input/reading-habit-dataset/BigML_Dataset_5f50a62795a9306aa200003e.csv")
# Due to the considerable number of samples, we will visualise only the first 5 rows of the dataset
df.head()
```

```
# overview of the data
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2832 entries, 0 to 2831
Data columns (total 14 columns):
 #   Column                                                Non-Null Count  Dtype
---  ------                                                --------------  -----
 0   Age                                                   2832 non-null   int64
 1   Sex                                                   2832 non-null   object
 2   Race                                                  2832 non-null   object
 3   Marital status?                                       2832 non-null   object
 4   Education                                             2832 non-null   object
 5   Employement                                           2832 non-null   object
 6   Incomes                                               2832 non-null   object
 7   How many books did you read during last 12months?     2832 non-null   int64
 8   Read any printed books during last 12months?          2442 non-null   object
 9   Read any audiobooks during last 12months?             2442 non-null   object
 10  Read any e-books during last 12months?                2442 non-null   object
 11  Last book you read, you...                              2442 non-null   object
 12  Do you happen to read any daily news or newspapers?   2832 non-null   object
 13  Do you happen to read any magazines or journals?      2832 non-null   object
dtypes: int64(2), object(12)
memory usage: 309.9+ KB
```

Examining Individual Variables

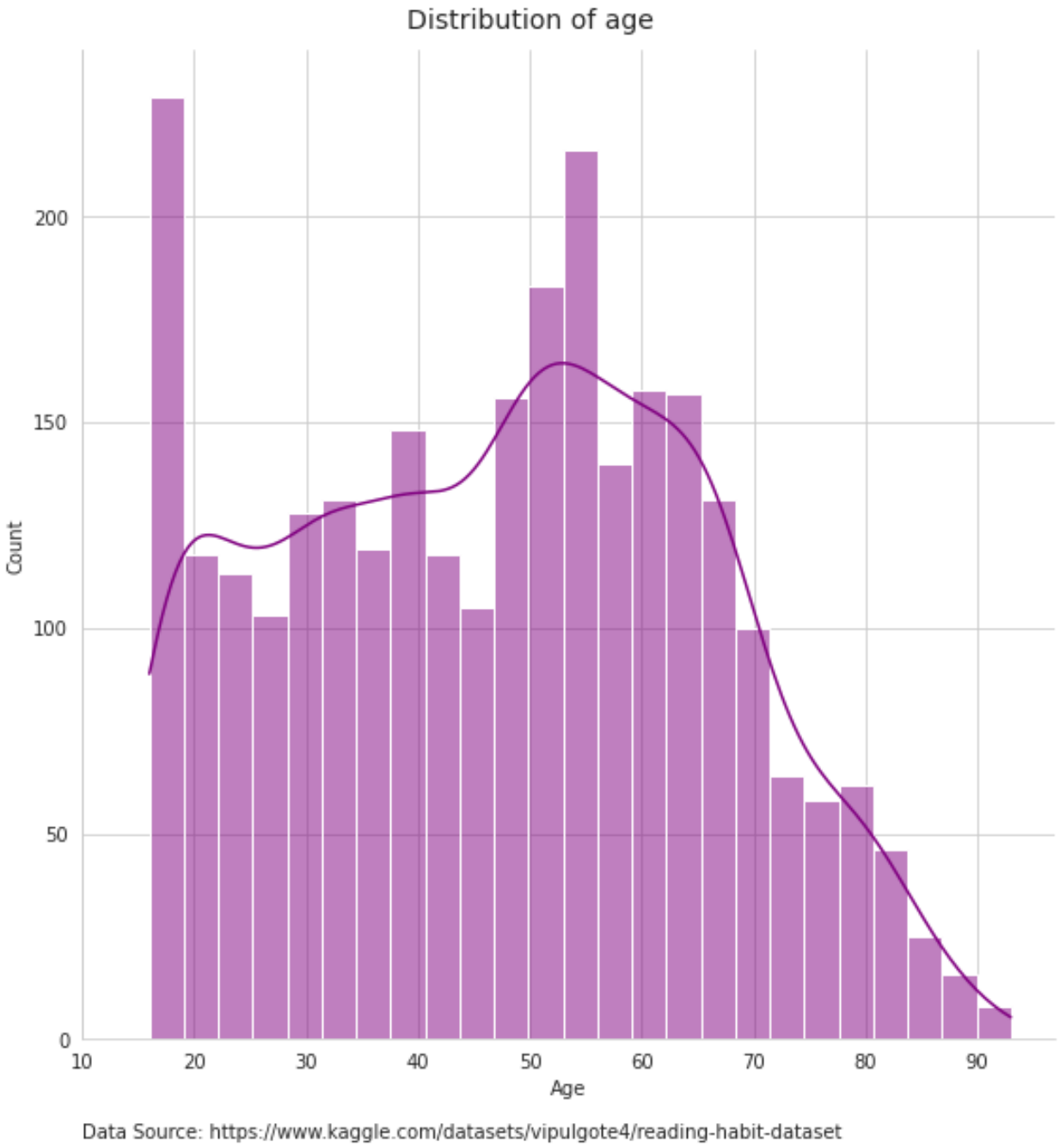
Age

The participants' age ranged from 16 to 93 years old. The average participant was around 47 years old.

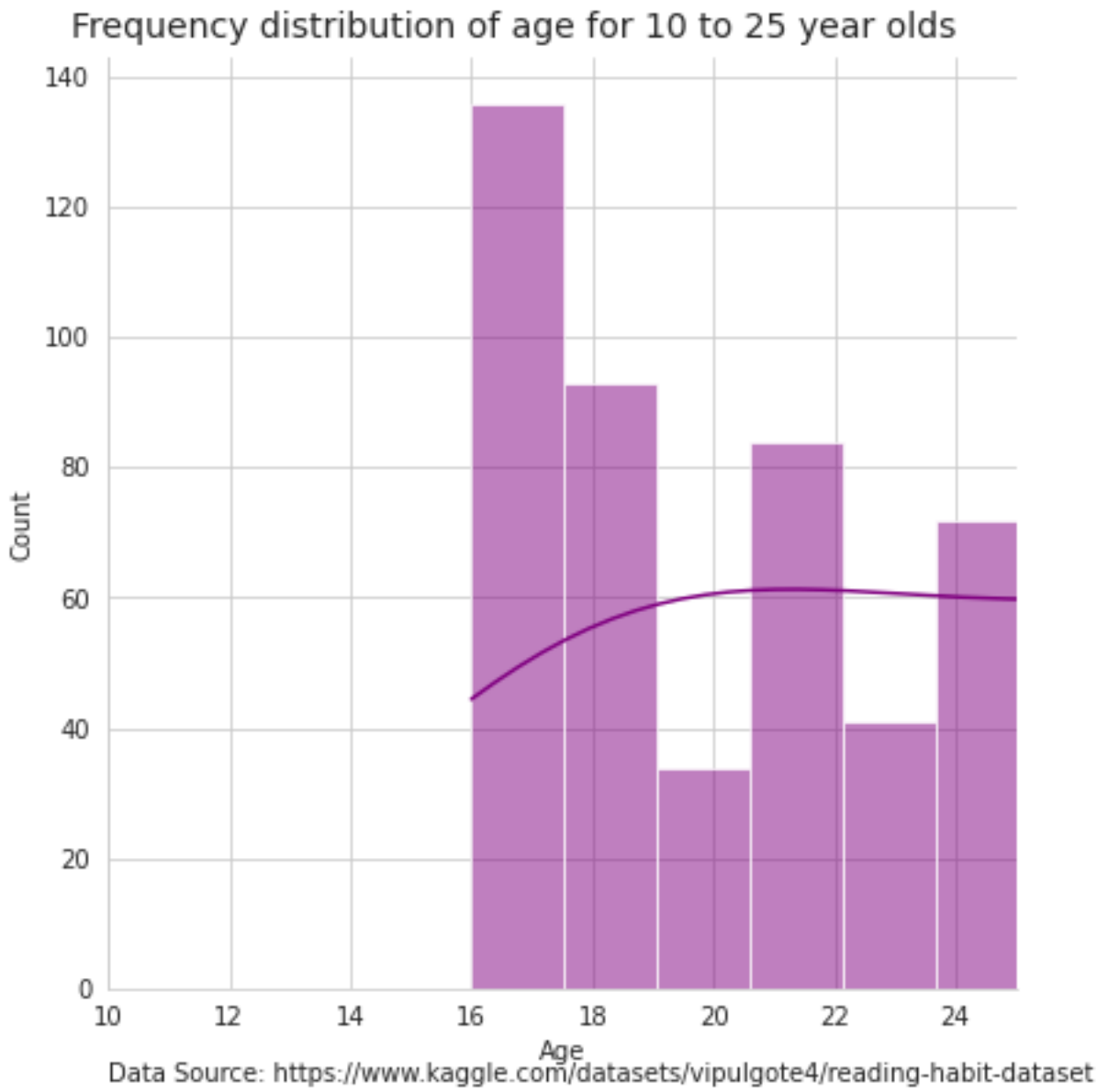
```
# Let's check the stats of the 'Age' variable
print("Age stats")
df["Age"].describe().rename_axis("Stat").reset_index(name = "value").set_index("Stat")
```

```
Age stats
```

```
sns.displot(x= "Age", data = df, bins= 25, kde = True, height = 8, color = "purple")

plt.xlim(10,)
plt.suptitle("Distribution of age", y=1.02, fontsize = 14)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), x
plt.show()
```




```
sns.displot(x= "Age", data = df, bins= 50, kde = True, height = 6, color = "blue")


plt.xlim(10,25)
plt.suptitle("Frequency distribution of age for 10 to 25 year olds", y=1.02, fontsize = 14)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), 
plt.show()
```




```
print("There are", sum(df["Age"]== 16), "16 year olds in this dataset.")
print("This constitutes",round((sum(df["Age"]== 16)/df.shape[0])*100, 2), "% of the entire dataset")
```

```
There are 68 16 year olds in this dataset.
This is 2.4 % of the entire dataset
```

Sex

This variable does not take into account non-binary genders and has to be answered by all participants, so there are only two possible outcomes. It seems most of the participants were female.

```
# view distribution of genders
print("Sex stats")
df["Sex"].value_counts()
```

```
Sex stats

Female    1479
Male      1353
Name: Sex, dtype: int64
```

```
print(round((df[df["Sex"]=="Female"].shape[0]/df.shape[0])*100,2),"% of the participents were female")
```

```
52.22 % of the participants were female
```

Race

This variable provide us with more nuanced answers compared to the previous one, as participants were allowed not to answer it, although the dataset summarises some of the non-answers with "don't know", which is quite perplexing.

```
# Let's check the main stats regarding the race of the participants
print("Race stats")
df["Race"].value_counts().rename_axis("Stat").reset_index(name = "value").set_index("Stat")
```

```
Race stats
```

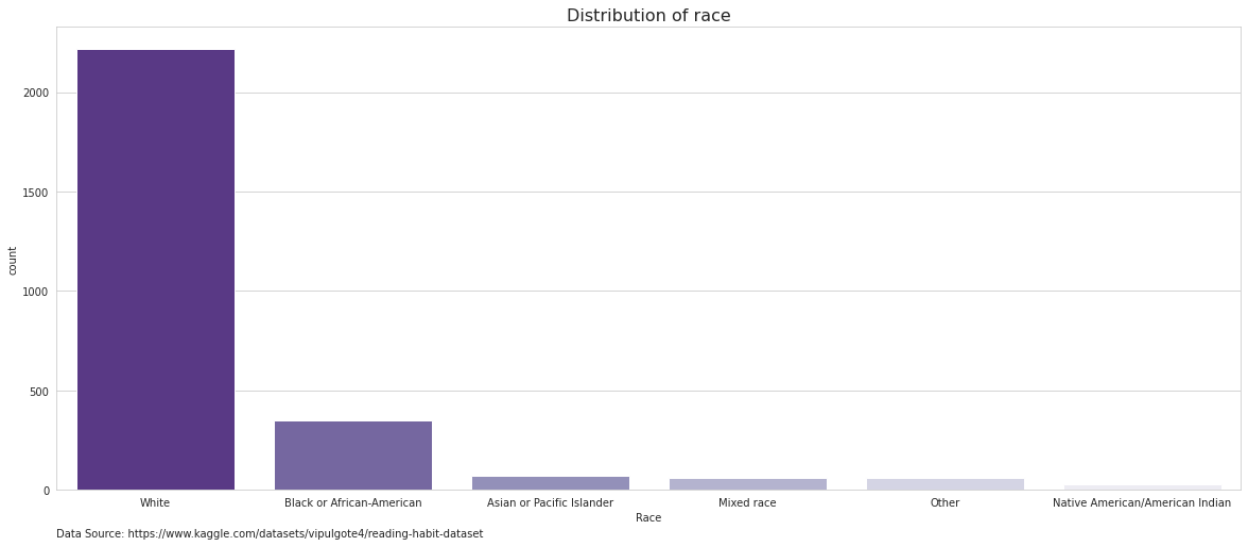
```
print("Of all participants", sum(df["Race"]=="White"), "are white.")
print("This means",round((sum(df["Race"]=="White")/df.shape[0])*100, 2), "% of the entire dataset is co
```

Of all participants, 2217 are white. This means 78.28 % of the entire dataset is made up of white individuals.

As mentioned earlier, the "I don't know" answers are quite perplexing. If I were to speculate, I'd suggest that these answers are due to American sensibility regarding one's ancestry(it is not uncommon for some citizens to do "DNA Ancestry tests" and analyse where their ancestors supposedly come from) or that they belong to individuals of mixed race unaware of their complete background.

```
# no matter the reasoning, we need more accuracy. Let's take care of the missing values
df["Race"].replace(("Don't know","Refused"), np.NaN, inplace = True)
```

```
plt.figure(figsize=(20,8))
sns.countplot(x= "Race", data = df, palette = "Purples_r", order = df["Race"].value_counts().index )

plt.title("Distribution of race", fontsize = 16)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), 
plt.show()
```



Marital status

```
# revise column name
df.rename(columns = {"Marital status?":"Marital status"}, inplace = True)
```

For this variable it is not so strange to see some "don't know" answers. Some people don't like to "label" their relationships. However, it seems there's something else missing...

```
# let's who is single and who isn't
print("Marital status stats")
df['Marital status'].value_counts().rename_axis("Response").reset_index(name = "count").set_index("Resp
```

```
Marital status stats

So we have 17 individuals who do not know if they are in a relationship or not.
They account for 0.6 % of responses.
```

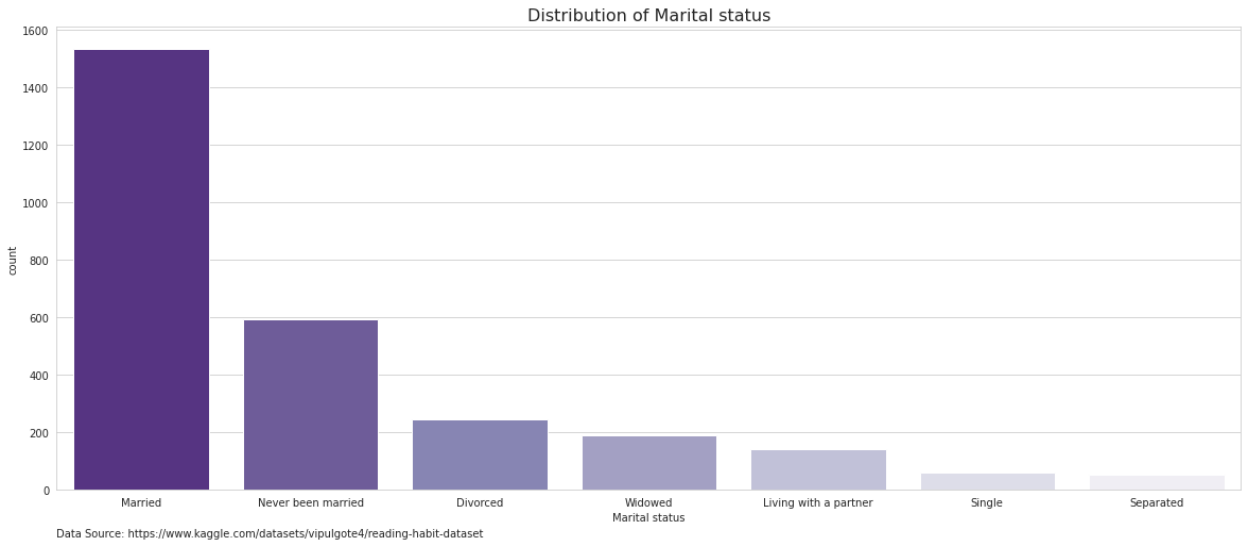
```
# Again, we need to be as accurate as possible
df["Marital status"].replace("Don't know", np.NaN, inplace = True)
```

So, most of those interviewed are married. Good for them.

... but wait, what about people who are in relationship but do NOT live with their partner? Given how common a situation this is, the fact that they were not taken into account is a glaring issue.

We have no way of knowing how many of the participants belong to this category, nor do we know what answer the gave. This severely undermines the accuracy of the survey.

```
plt.figure(figsize=(20,8))
sns.countplot(x = "Marital status" , data = df, order = df["Marital status"].value_counts().index, pale

plt.title("Distribution of Marital status", fontsize = 16)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), 
plt.show()
```



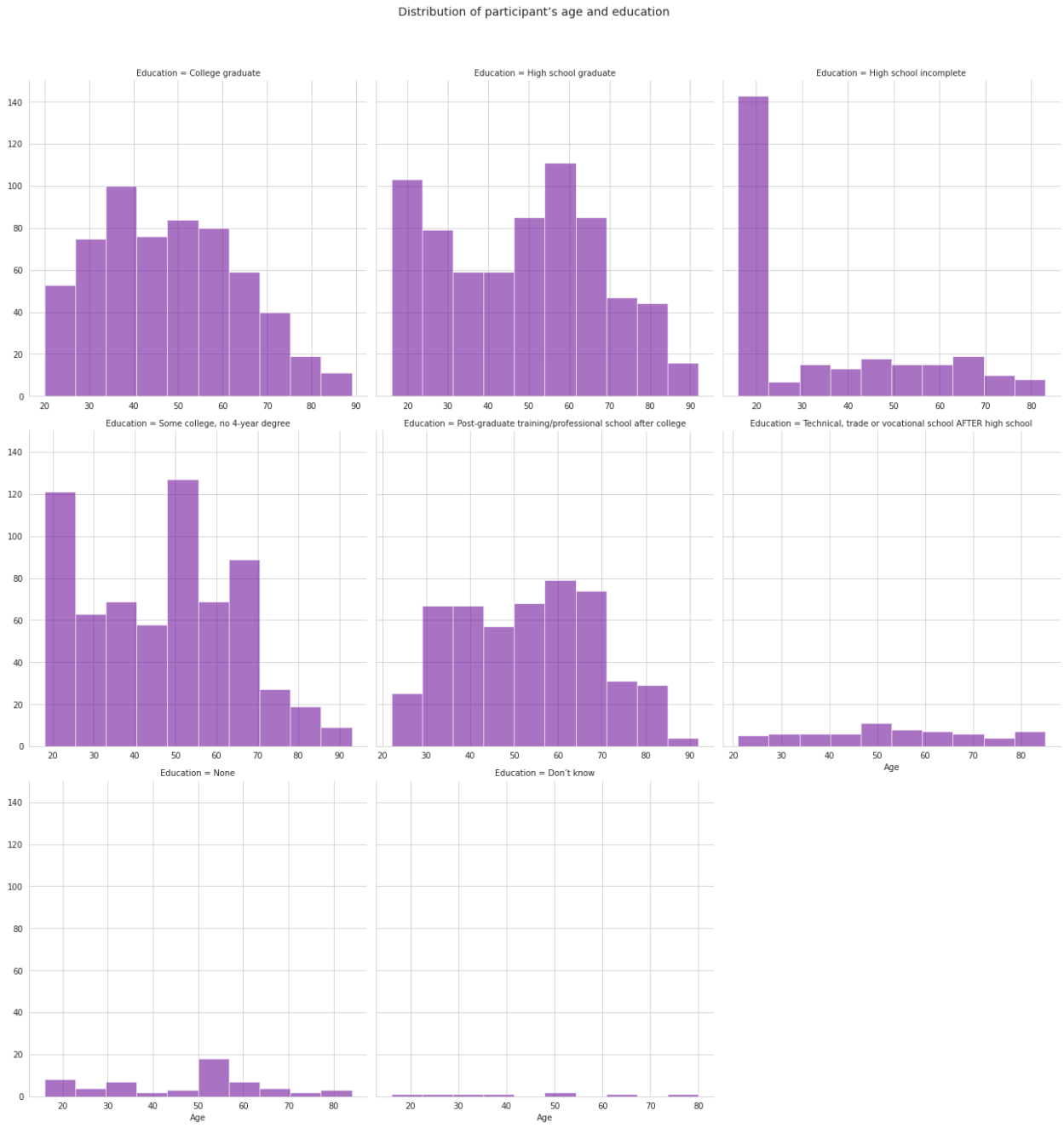
Education

```
# now it's time to check education
print("Education stats")
df["Education"].value_counts().rename_axis("Response").reset_index(name = "value").set_index("Response")
```

```
Education stats
```

```
97.95% of the participants indicated they had received education of any kind.
```

```
g = sns.FacetGrid(df, col = "Education", sharex=False, col_wrap = 3, height = 6, )
g.map(plt.hist, "Age", color = "#70149c", alpha = 0.6)
plt.suptitle("Distribution of participant's age and education", y=1.05, fontsize = 14)
plt.show()
```



After checking the data, we notice another flaw:

According to their age, several participants had to have been high school students at the time. However, there isn't a choice suitable for their condition, as the surveys assumes people have abandoned high school or completed it with success, with no middle choices.

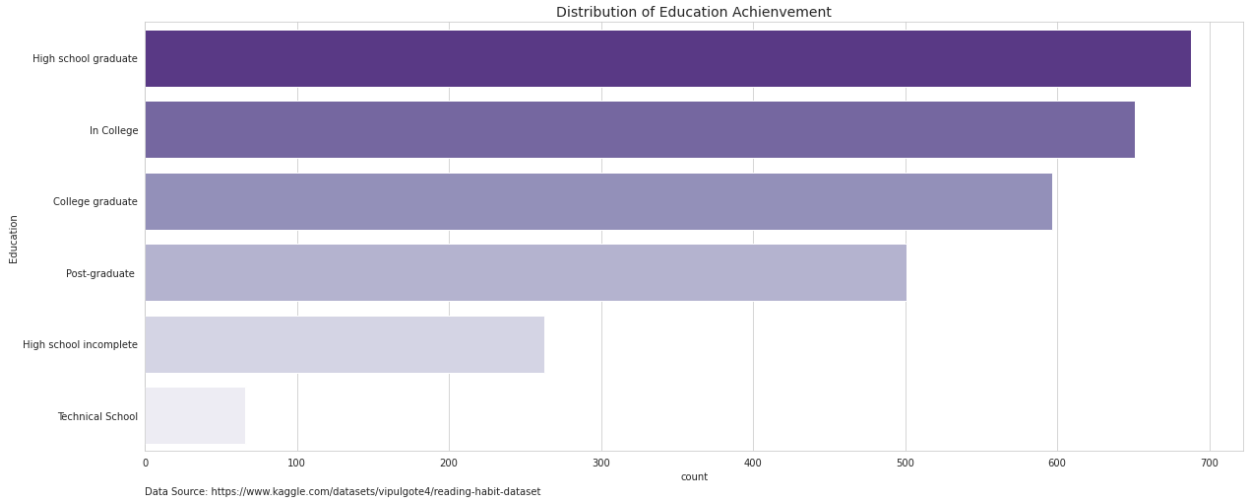
Thus, how are we to take into account the missing values? Afterall, high school students may have picked the "don't know" or "none option".

Alongside revising "none" and "don't know" to null values, it may be wise to revise the category names to a more aesthetic name, whilst attempting to ensure the responses remain accurate.

```
# Once more, let's try to get the best accuracy possible
df["Education"].replace(("None", "Don't know"), np.NaN, inplace = True)
df["Education"].replace("Technical, trade or vocational school AFTER high school", "Technical School", i
df["Education"].replace("Post-graduate training/professional school after college", "Post-graduate ", in
df["Education"].replace("Some college, no 4-year degree", "In College", inplace = True)

print("Education distribution")
df["Education"].value_counts().rename_axis("Response").reset_index(name = "count").set_index("Response")
```

```
Education distribution
```

```
plt.figure(figsize=(20,8))
sns.countplot(y = "Education", data = df, order = df["Education"].value_counts().index, palette = "Blues

plt.title("Distribution of Education Achienvement", fontsize = 14)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), i
plt.show()
```



Employement

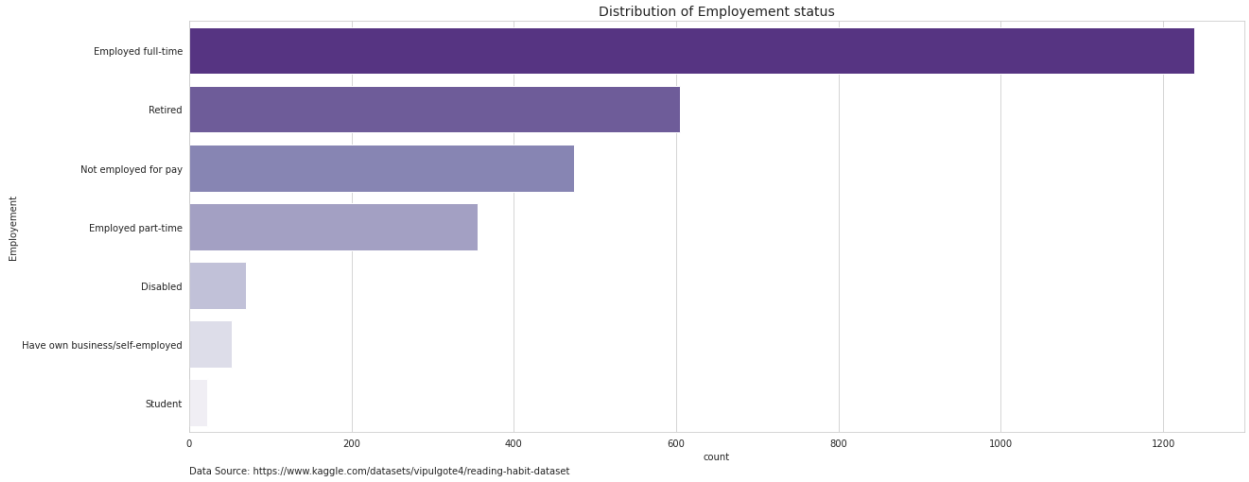
Finally, a variable that is mostly without issues. There are null values here as well, but there isn't much we can do about them.

```
# Who's employed? and who isn't?
print("Employement distribution")
df["Employement"].value_counts().rename_axis("Response").reset_index(name = "count").set_index("Respons
```

```
Employement distribution
```

```
# once again unto the breach
df["Employement"].replace("Other", np.NaN, inplace = True)
```

```
plt.figure(figsize=(20,8))
sns.countplot(y = "Employement", data = df, order = df["Employement"].value_counts().index, palette = "
plt.title("Distribution of Employement status", fontsize = 14)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), 
plt.show()
```



Incomes

Income is yet another weak variable, but not because of dubious answers. This time, the problem is how wide the difference in dollars is between categories. This also cause the problem of grouping up people who are "close in wealth".

```
# Time to see the stats for income, and how it is distributed
print("Incomes distribution")
df["Incomes"].value_counts().rename_axis("Response").reset_index(name = "count").set_index("Response")
```

```
Incomes distribution
```

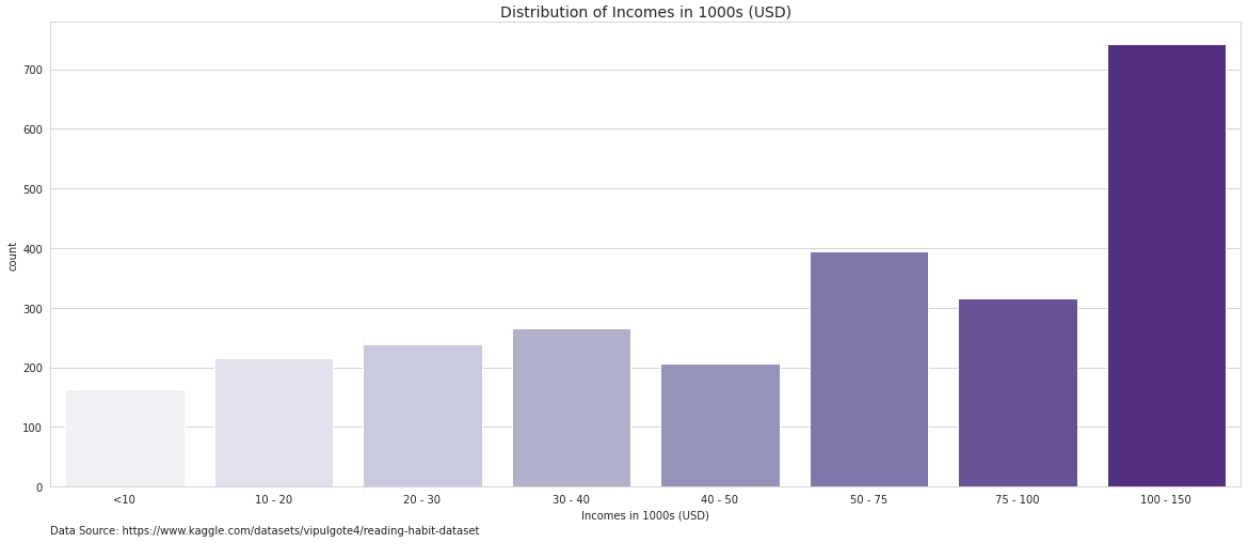
```
# We might as well change the format a bit

df["Incomes"].replace("$100,000 to under $150,000", "100 - 150", inplace = True)
df["Incomes"].replace("$50,000 to under $75,000", '50 - 75', inplace = True)
df["Incomes"].replace("$75,000 to under $100,000", '75 - 100', inplace = True)
df["Incomes"].replace("Refused", np.NaN, inplace = True)

df["Incomes"].replace("$30,000 to under $40,000", "30 - 40", inplace = True)
df["Incomes"].replace("$20,000 to under $30,000", "20 - 30", inplace = True)
df["Incomes"].replace("$10,000 to under $20,000", "10 - 20", inplace = True)
df["Incomes"].replace("9$100,000 to under $150,000", "100 - 150", inplace = True)
df["Incomes"].replace("$40,000 to under $50,000", "40 - 50", inplace = True)
df["Incomes"].replace("Less than $10,000", "<10", inplace = True)
```

```
df.rename(columns = {'Incomes':'Incomes in 1000s (USD)'}, inplace = True)
```

```
print("Income distribution")
df["Incomes in 1000s (USD)"].value_counts().rename_axis("Response").reset_index(name = "count").set_ind
```

```
Income distribution
```

```
plt.figure(figsize=(20,8))
sns.countplot(x = "Incomes in 1000s (USD)", data = df, order = ["<10", "10 - 20", "20 - 30",
                                                                "30 - 40","40 - 50","50 - 75",
                                                                "75 - 100", "100 - 150" ], palette = "Reds")

plt.title("Distribution of Incomes in 1000s (USD)", fontsize = 14)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), x
plt.show()
```



No. of books read in the last 12 months

Now we can see about much people read or do not read. Let's see.

```
print("Most popular number of books read in the last 12months distribution")
df["No. of books read in the last 12months"].value_counts( ascending= False).rename_axis ("No. of books
```

```
Most popular number of books read in the last 12months distribution
```

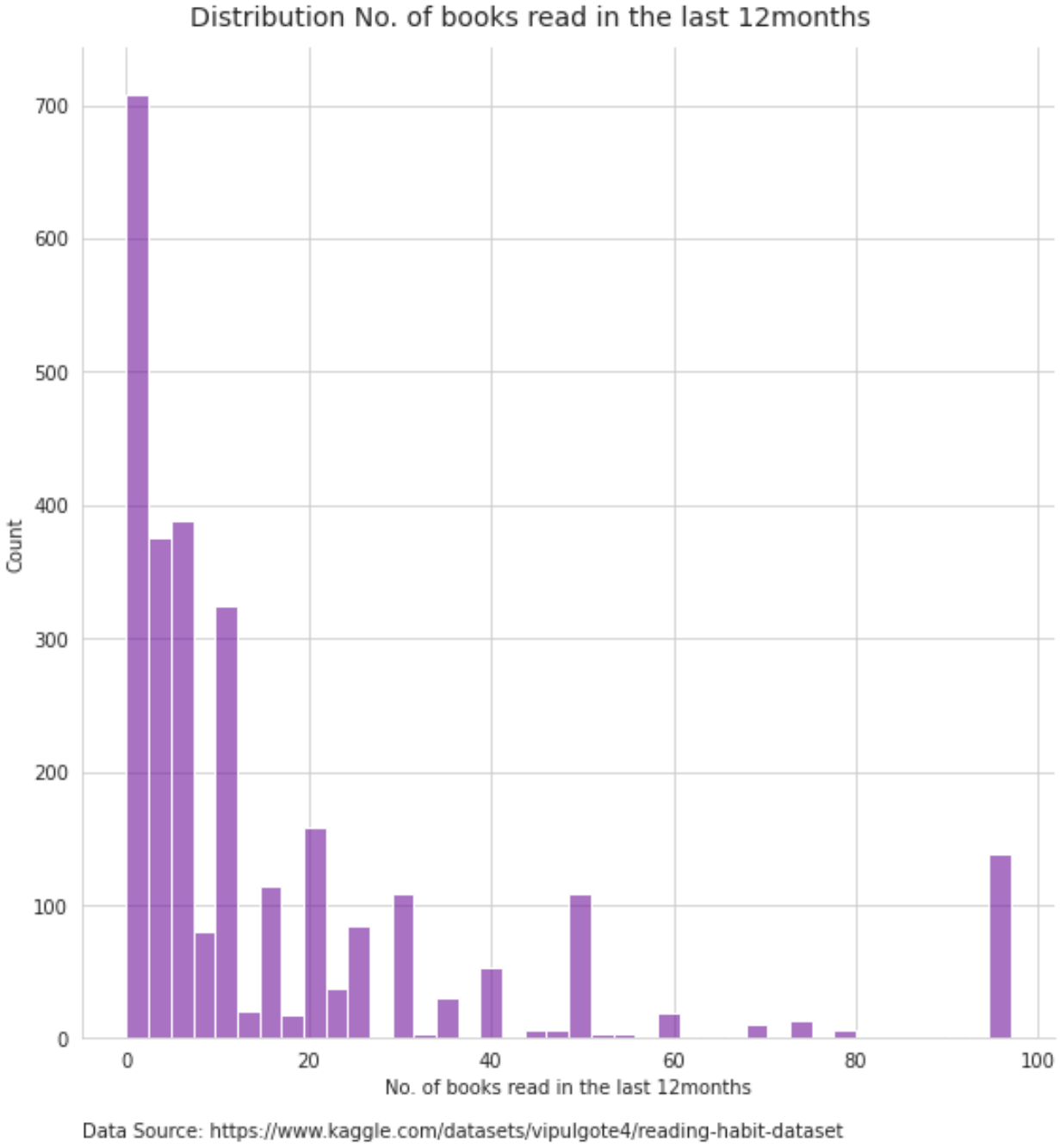
```
 print("No. of books read in last 12 months stats")
df["No. of books read in the last 12 months"].describe().rename_axis("stat").reset_index(name = "value"
```

```
No. of books read in last 12 months stats
```

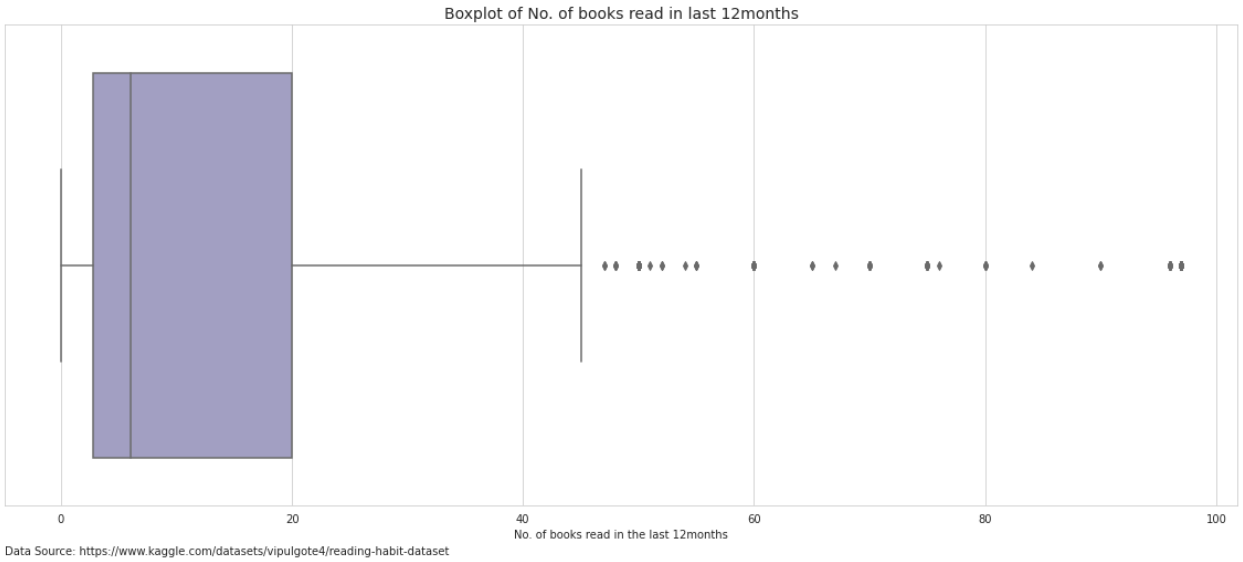
```
print(round((df[df["No. of books read in the last 12months"]<5].shape[0]/df.shape[0])*100,2),"% of parti
```

```
38.24 % of participants read less than 5 books
```

```
sns.displot( x = "No. of books read in the last 12months", data = df, height = 8, color = "#70149c", alp

plt.suptitle("Distribution No. of books read in the last 12months", y = 1.02, fontsize = 14)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), x
plt.show()
```



```
plt.figure(figsize=(20,8))
sns.boxplot(x= "No. of books read in the last 12months", data = df, palette = "Purples")

plt.title("Boxplot of No. of books read in last 12months", fontsize = 14)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), x
plt.show()
```



Printed books read in the last 12 months

```
df.rename(columns = {'Read any printed books during last 12months?':"Printed books in the last 12ms"}, i
```

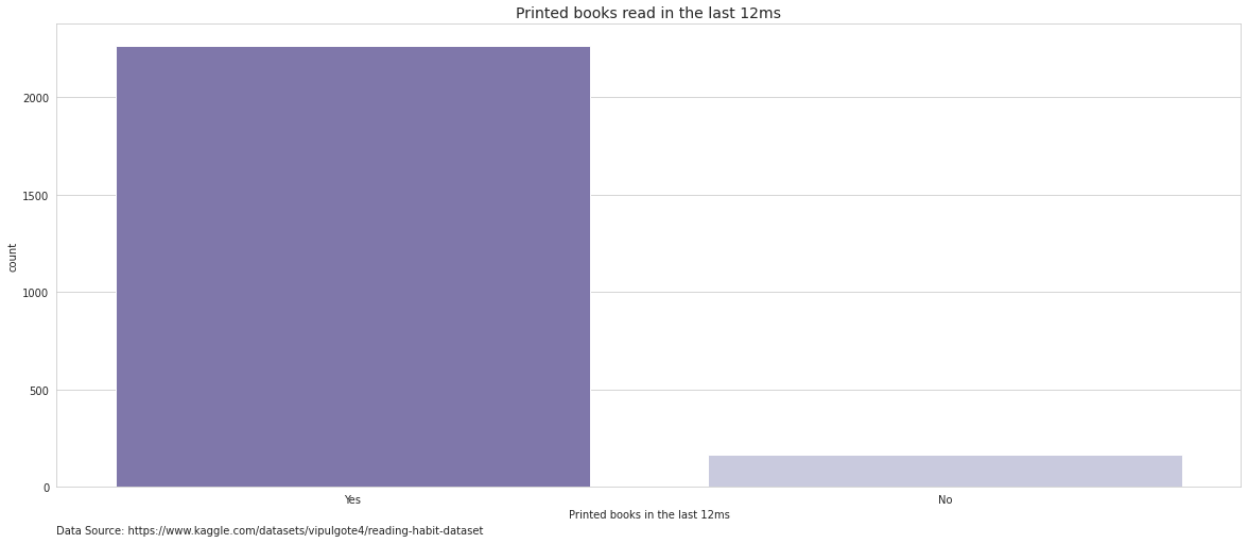
It seems most people have read at least one printed book in the last year.

```
print("Printed books read in the last 12months distribution")
df["Printed books in the last 12ms"].value_counts().rename_axis ("Response").reset_index(name = "Count"
```

```
Printed books read in the last 12 months distribution
```

```
df["Printed books in the last 12ms"].replace("Don't know", np.NaN, inplace= True)
```

```
plt.figure(figsize=(20,8))
sns.countplot(x= "Printed books in the last 12ms", data = df, palette = "Purples_r", order = ["Yes", "N


plt.title("Printed books read in the last 12ms", fontsize = 14)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), x
plt.show()
```



Audio and eBooks

```
df.rename(columns = {"Read any audiobooks during last 12months?":"Audiobooks in last 12ms"}, inplace = T
```

```
# Let's check this variable
print("Audiobooks in last 12months distribution")
df["Audiobooks in last 12ms"].value_counts().rename_axis("Response").reset_index(name = "Count").set_in
```

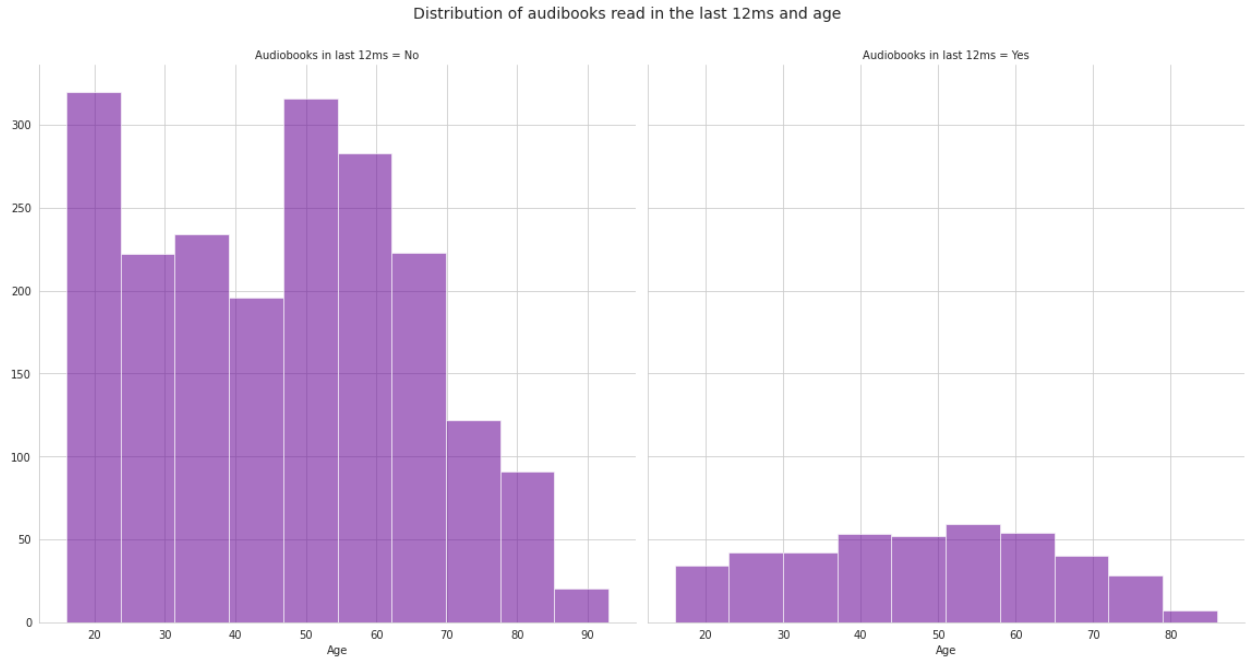
```
Audiobooks in last 12months distribution
```

```
# let's take care of the missing values
df["Audiobooks in last 12ms"].replace("Don't know", np.NaN, inplace = True)
```

```
readers = df[df["No. of books read in the last 12months"]>0]

non_readers = df[df["No. of books read in the last 12months"]<=0]
print(len(non_readers["No. of books read in the last 12months"]), "of individuals read no books in the l
```

```
390 of individuals read no books in the last 12months
```

```
# create graph
g = sns.FacetGrid(readers, col = "Audiobooks in last 12ms", sharex=False,height = 8)
g.map(plt.hist, "Age", color = "#70149c", alpha = 0.6)

# format graph
plt.suptitle("Distribution of audibooks read in the last 12ms and age", y=1.05, fontsize = 14)
plt.show()
print("Note: the data here has been resampled to only include individuals who have read at least one boo
```



Note: the data here has been resampled to only include individuals who have read at least one book.

```
# renaming the variable
df.rename(columns = {"Read any e-books during last 12months?":"E-Books during last 12ms"}, inplace = Tru
```

```
print("E-Books during last 12ms distribution")
df["E-Books during last 12ms"].value_counts().rename_axis("Response").reset_index(name = "Count").set_in
```

```
E-Books during last 12ms distribution
```

```
# revise questionable responses to null/na
df["E-Books during last 12ms"].replace("Don't know", np.NaN, inplace = True)
```

Surprisingly, it seems audiobooks and e-books are not particularly popular.

Method of obtaining previous book

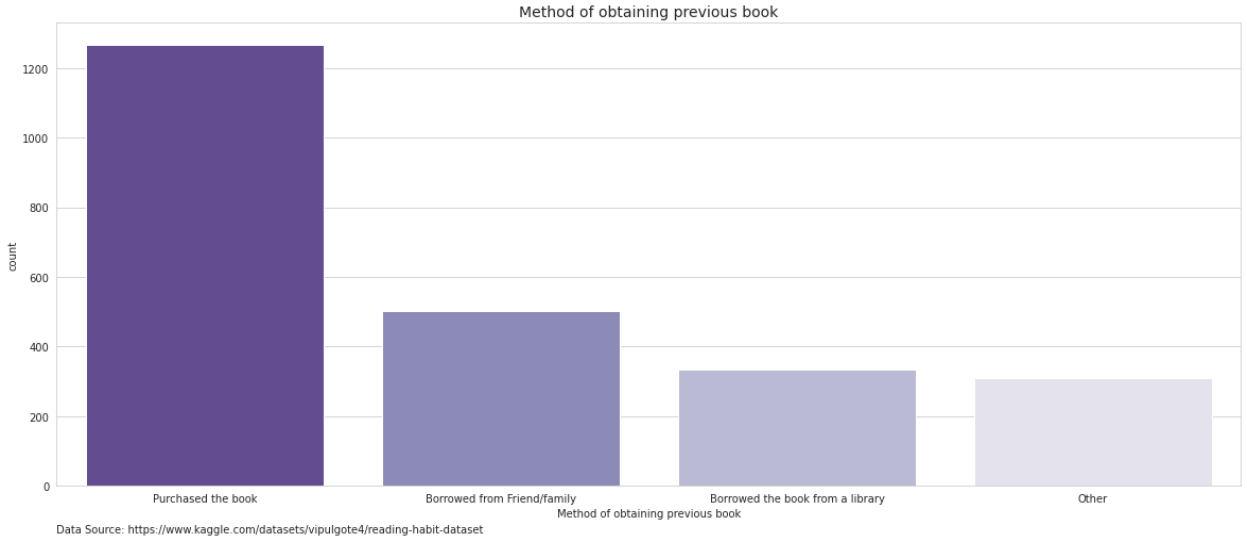
```
# this one variable has a particularly awkward name
df.rename(columns = {"Last book you read, you...":"Method of obtaining previous book"}, inplace = True)
```

```
print("Method of obtaining previous book distribution")
df["Method of obtaining previous book"].value_counts().rename_axis("Response").reset_index(name = "Count
```

```
Method of obtaining previous book distribution
```

```
# we need to fix some mistaken values, by treating them as missing values
df["Method of obtaining previous book"].replace(["8","9"], np.NaN, inplace = True)
df["Method of obtaining previous book"].replace("Borrowed the book from a friend or family member", "Bo
df["Method of obtaining previous book"].replace("Got the book some other way", "Other", inplace = True)
```

```
plt.figure(figsize=(20,8))

sns.countplot(x = "Method of obtaining previous book", data = df, palette = "Reds_r")
plt.title("Method of obtaining previous book", fontsize = 14)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), 
plt.show()
```



Newspapers & Magazines

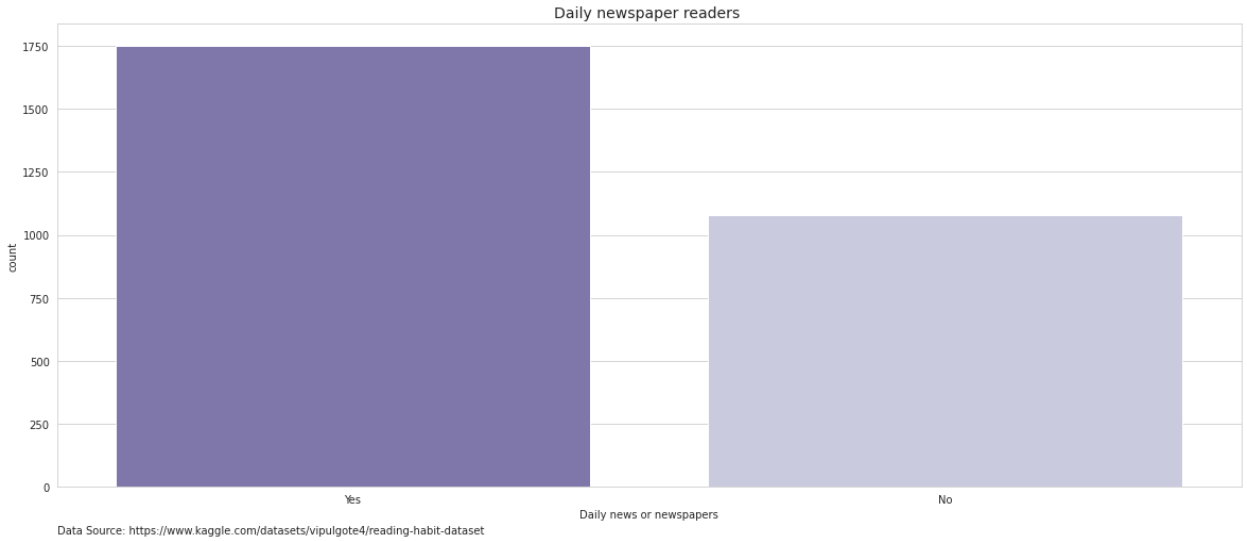
On paper, this variable is one of the secure ones, and if we check the stats we'll see that most of the participants were read newspapers or magazines.

```
print("Daily news or newspapers distribution")
df["Daily news or newspapers"].value_counts().rename_axis("Response").reset_index(name = "Count").set_i
```

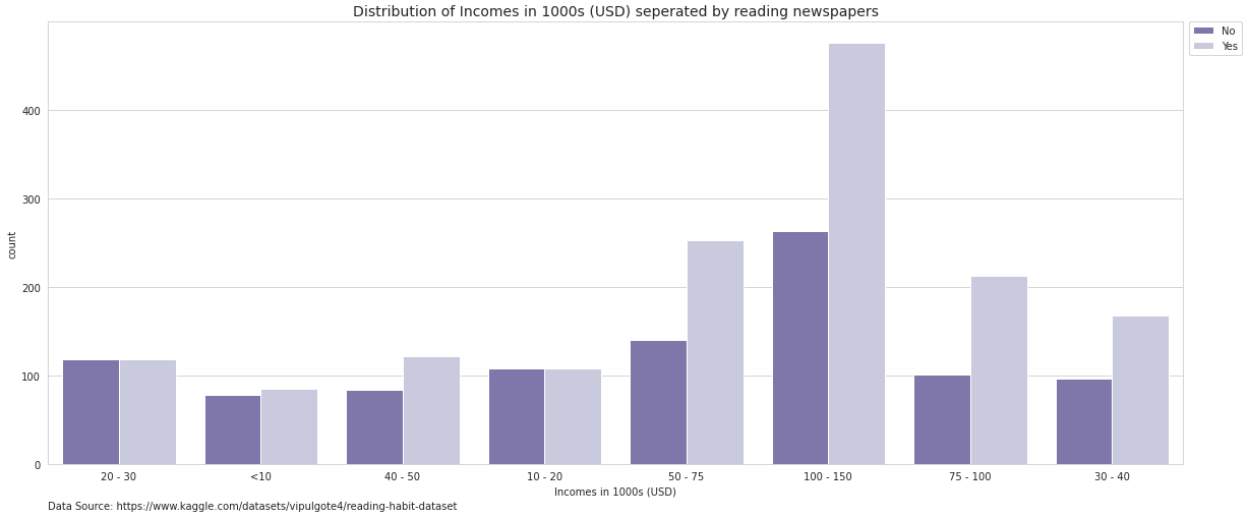
```
Daily news or newspapers distribution
```

```
df["Daily news or newspapers"].replace("Don't know", np.NaN, inplace = True)
```

```
plt.figure(figsize=(20,8))

sns.countplot(x = "Daily news or newspapers", data = df, palette = "Purples_r", order = ["Yes", "No"])
plt.title("Daily newspaper readers ", fontsize = 14)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), 
plt.show()
```




```
plt.figure(figsize = (20,8))
sns.countplot(x = "Incomes in 1000s (USD)", data = df, hue = "Daily news or newspapers", palette = "Purp
plt.title("Distribution of Incomes in 1000s (USD) seperated by reading newspapers", fontsize = 14)
plt.legend(bbox_to_anchor=(1.005, 1), loc='upper left', borderaxespad=0)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), x
plt.show()
```



Magazines or Journals

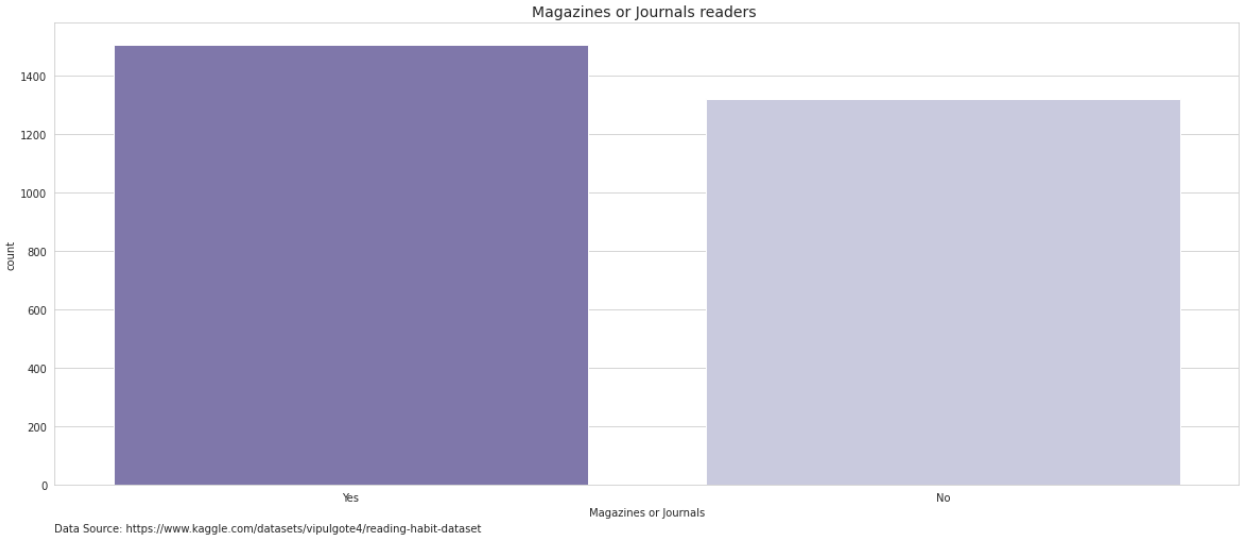
This variable is a follow-up of the last one. By checking the stats, we see that the habits of the partcipants are more or less "balanced".

```
 print("Magazines or Journals distribution")
df["Magazines or Journals"].value_counts().rename_axis("Response").reset_index(name = "Count").set_index
```

```
Magazines or Journals distribution
```

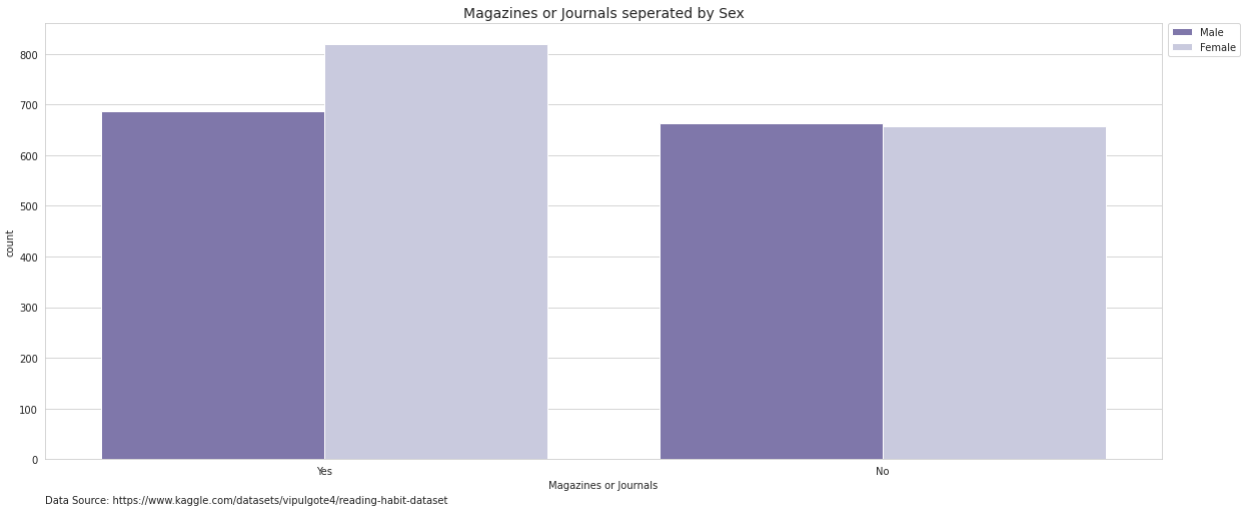
```
df["Magazines or Journals"].replace("Don't know", np.NaN, inplace = True)
```

```
plt.figure(figsize = (20,8))
sns.countplot(x = "Magazines or Journals", data = df, order = ["Yes", "No"], palette = "Purples_r",)
plt.title("Magazines or Journals readers ", fontsize = 14)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), x
plt.show()
```



Magazines are more popular with females . Remember that the majority of the participants who are female, and most are reporting that they had read a magazine or journal over the past 12months.

```
plt.figure(figsize = (20,8))
sns.countplot(x = "Magazines or Journals", data = df, hue = "Sex", palette = "Purples_r",)
plt.title("Magazines or Journals seperated by Sex", fontsize = 14)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), x
plt.legend(bbox_to_anchor=(1.005, 1), loc='upper left', borderaxespad=0)

plt.show()
```



Insights and possible improvements

First of all, let's check the missing values we got:

```
print("Revised dataset - number of null values")
df.isna().sum().rename_axis("Variable").reset_index(name = "No. of Missing values").set_index("Variable
```

```
Revised dataset - number of null values
```

Now, let's what the 'new' dataset looks like:

```
df.head()
```

So to sum up, we know the average participants were 45 men and 48 years old women. We know that the minimum age recorded was 16 years old, but we are not sure if that was actually the real minimum age. Considering the 'peak' of 16 years old among the young participant, it is very possible the survey looked for age ranges, and not the age itself.

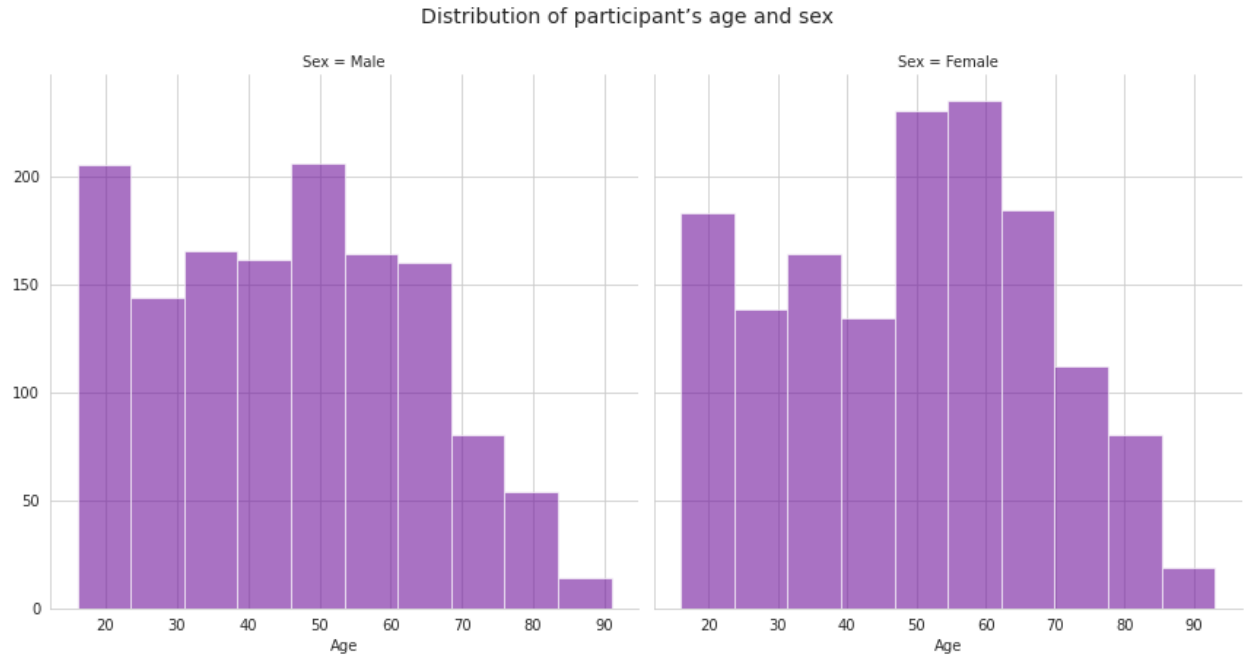
Most of the participants were white, and the participants who weren't tended to be younger. We also know that younger people tender to be singler while older people tended to be married.

However, we have seen that the dataset- and the survey- have several 'missing questions': many of them we discussed before, but there are others. For example, why were the participants not asked about the lenght or the genre of the books they read?

Another flaw is that the numbers themselves are strange: look at the abundance of 16 years old participants, for instance. It is possible that the survey asked for an age range?

Aside from the missing questions, the survey would benefit by not letting the participants skip some questions, especially given that anonymity is guaranteed.

```
g = sns.FacetGrid(df, col = "Sex", sharex=False, height = 6, )
g.map(plt.hist, "Age", color = "#70149c", alpha = 0.6)

plt.suptitle("Distribution of participant's age and sex",y=1.05, fontsize = 14);
```



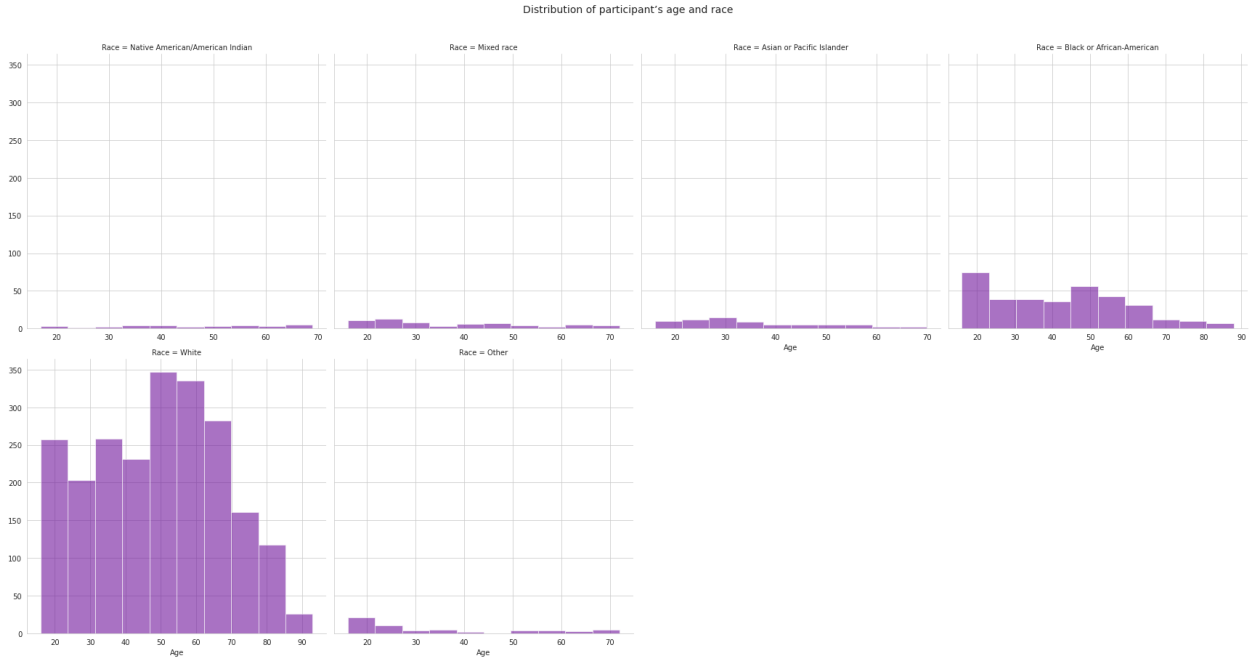
```
print("Age distribution of females")
df[df["Sex"]=="Female"]["Age"].describe().rename_axis("Stat").reset_index(name = "value").set_index("Sta
```

```
Age distribution of females
```

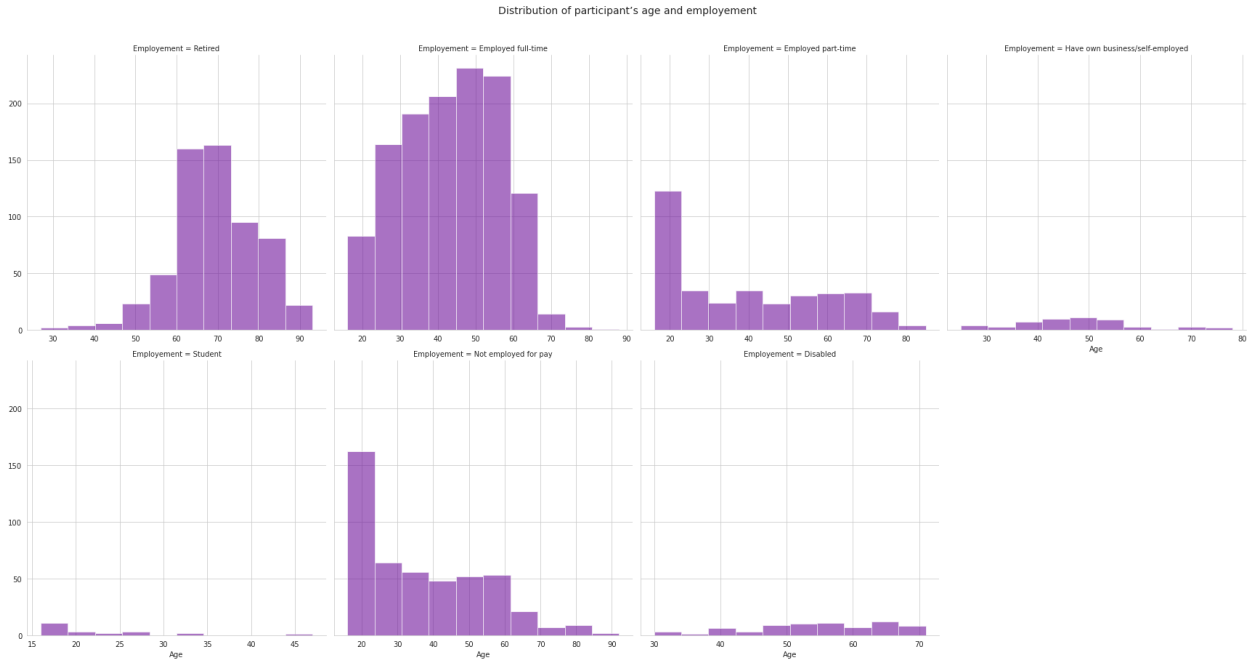
```
print("Age distribution of males")
df[df["Sex"]=="Male"]["Age"].describe().rename_axis("Stat").reset_index(name = "value").set_index("Stat
```

```
Age distribution of males
```

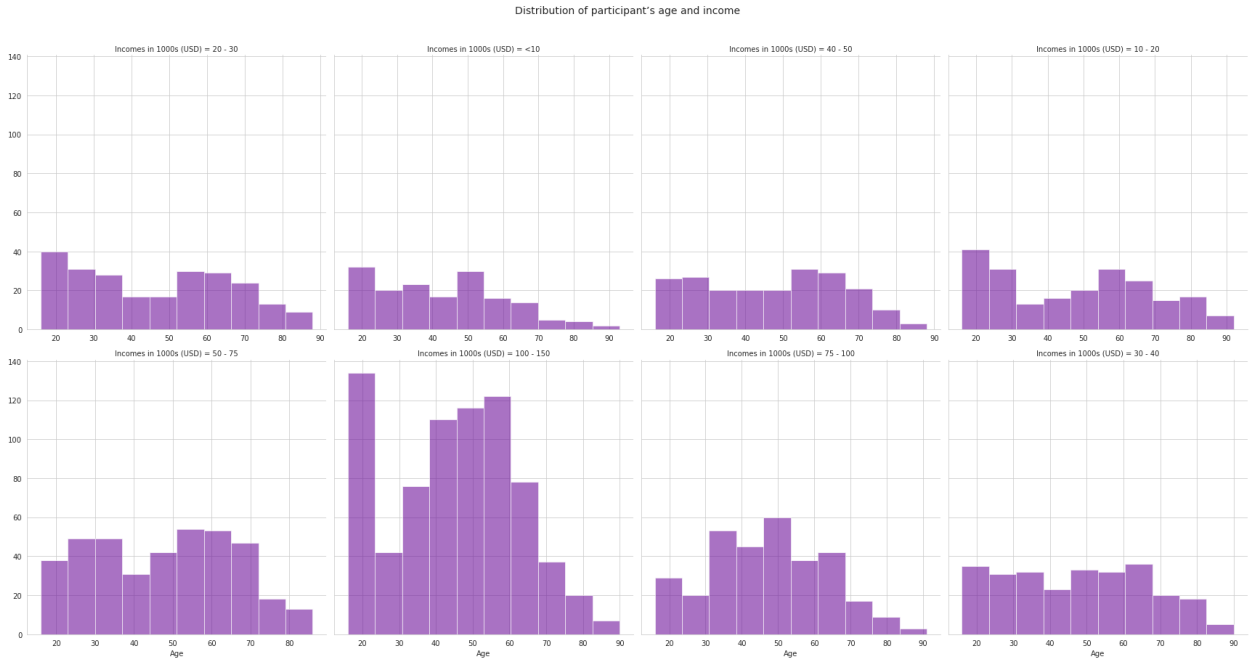
```
g = sns.FacetGrid(df, col = "Race", sharex=False, height = 6, col_wrap = 4)
g.map(plt.hist, "Age", color = "#70149c", alpha = 0.6)

plt.suptitle("Distribution of participant's age and race", y=1.05, fontsize = 14);
```



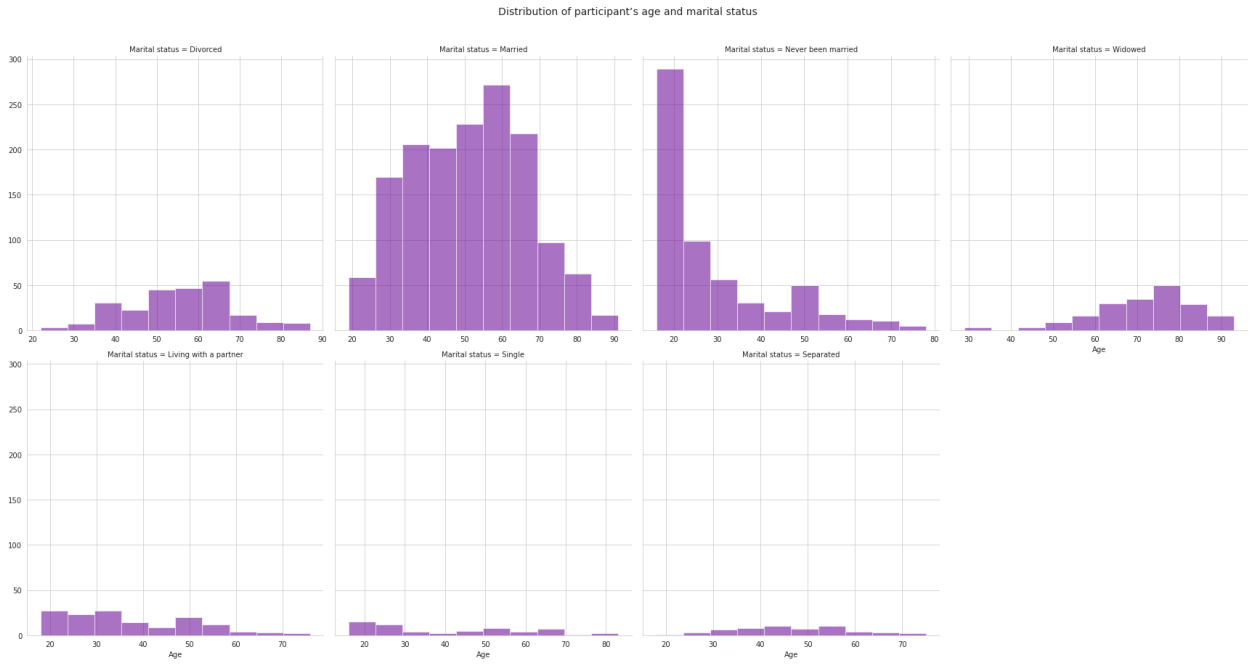

```
g = sns.FacetGrid(df, col = "Employement", sharex=False, height = 6, col_wrap = 4)
g.map(plt.hist, "Age", color = "#70149c", alpha = 0.6)

plt.suptitle("Distribution of participant's age and employement", y=1.05, fontsize = 14);
```




```
g = sns.FacetGrid(df, col = "Incomes in 1000s (USD)", sharex=False, height = 6, col_wrap=4)
g.map(plt.hist, "Age", color = "#70149c", alpha = 0.6)

plt.suptitle("Distribution of participant's age and income", y=1.05, fontsize = 14);
```

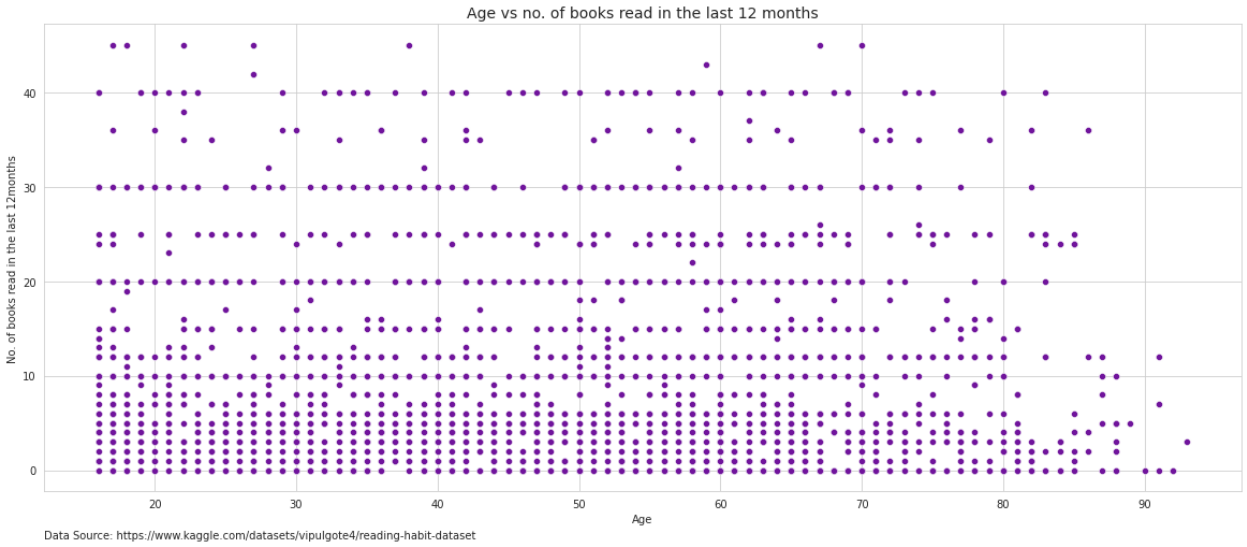


```
g = sns.FacetGrid(df, col = "Marital status", sharex=False, height = 6, col_wrap=4)
g.map(plt.hist, "Age", color = "#70149c", alpha = 0.6)

plt.suptitle("Distribution of participant's age and marital status", y=1.05, fontsize = 14);
```




```
ow#
Q1 =  df["No. of books read in the last 12months"].quantile(0.25)
Q3 = df["No. of books read in the last 12months"].quantile(0.75)
IQR = Q3 - Q1

df.drop(df[(df["No. of books read in the last 12months"] < Q1-1.5*IQR) | (df["No. of books read in the l
```

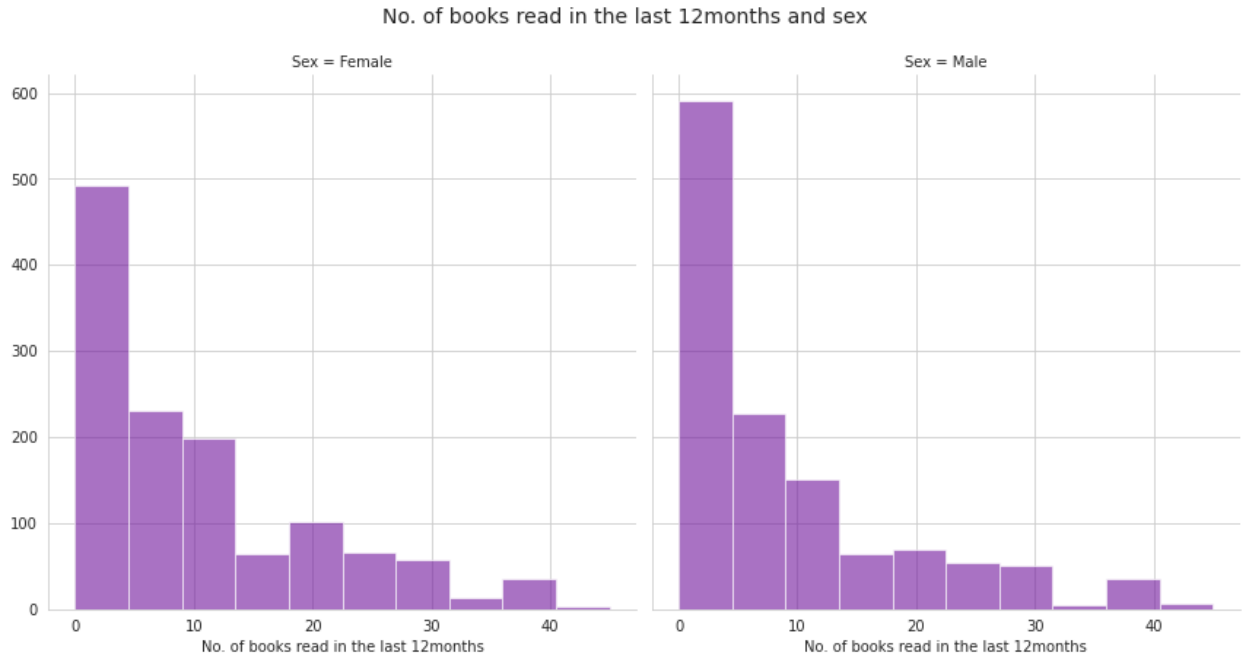
```
plt.figure(figsize=(20,8))
sns.scatterplot(x = "Age", y = "No. of books read in the last 12months", data = df, color = "#70149c",

plt.title("Age vs no. of books read in the last 12 months", fontsize = 14);
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), x
```




```
print("Sex and average number of books read recently")
df.groupby(["Sex"])["No. of books read in the last 12months"].mean().rename_axis("Sex").reset_index(name
```

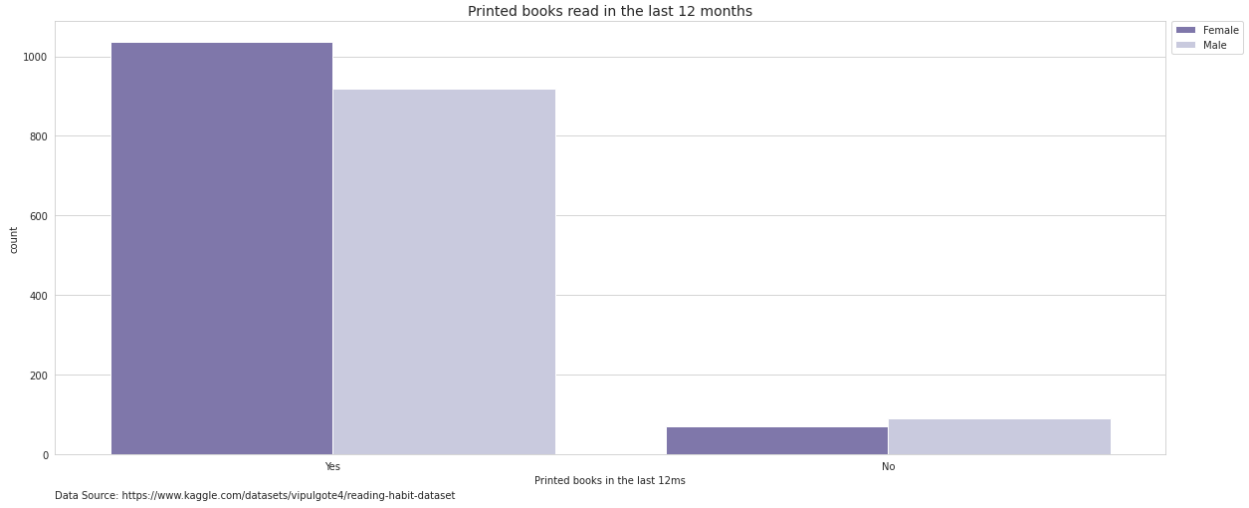
```
Sex and average number of books read recently
```

```
g = sns.FacetGrid(df, col = "Sex", sharex=False, height = 6)
g.map(plt.hist, "No. of books read in the last 12months", color = "#70149c", alpha = 0.6)

plt.suptitle("No. of books read in the last 12months and sex", y=1.05, fontsize = 14);
```

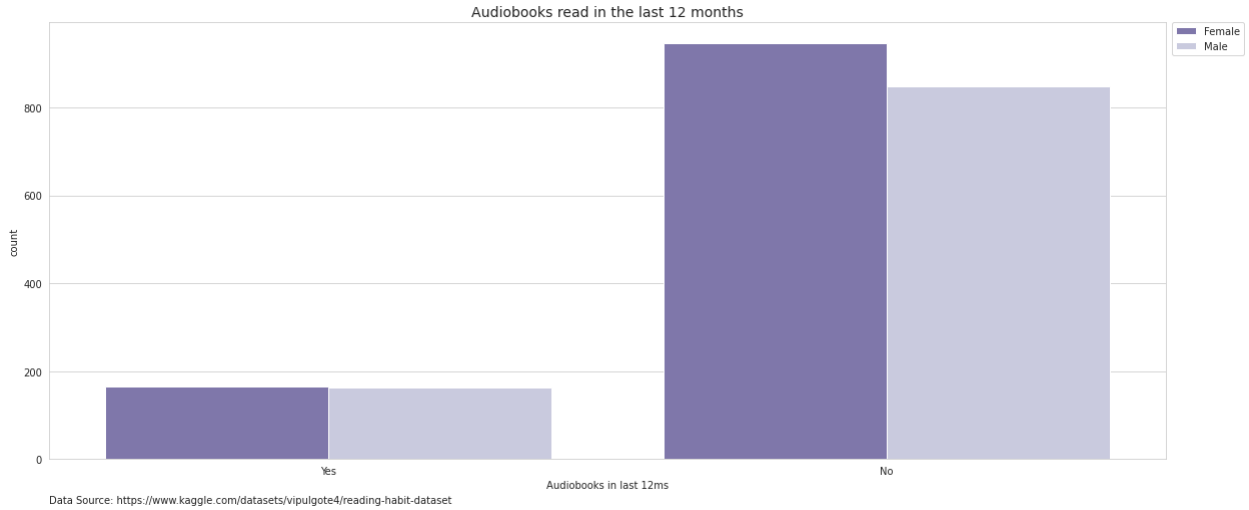



```
plt.figure(figsize=(20,8))
sns.countplot(x = "Printed books in the last 12ms", data = df, hue = "Sex",  order = ["Yes", "No"], hue
```

```
plt.title("Printed books read in the last 12 months", fontsize = 14)
plt.legend(bbox_to_anchor=(1.005, 1), loc='upper left', borderaxespad=0)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), x
plt.show()
```

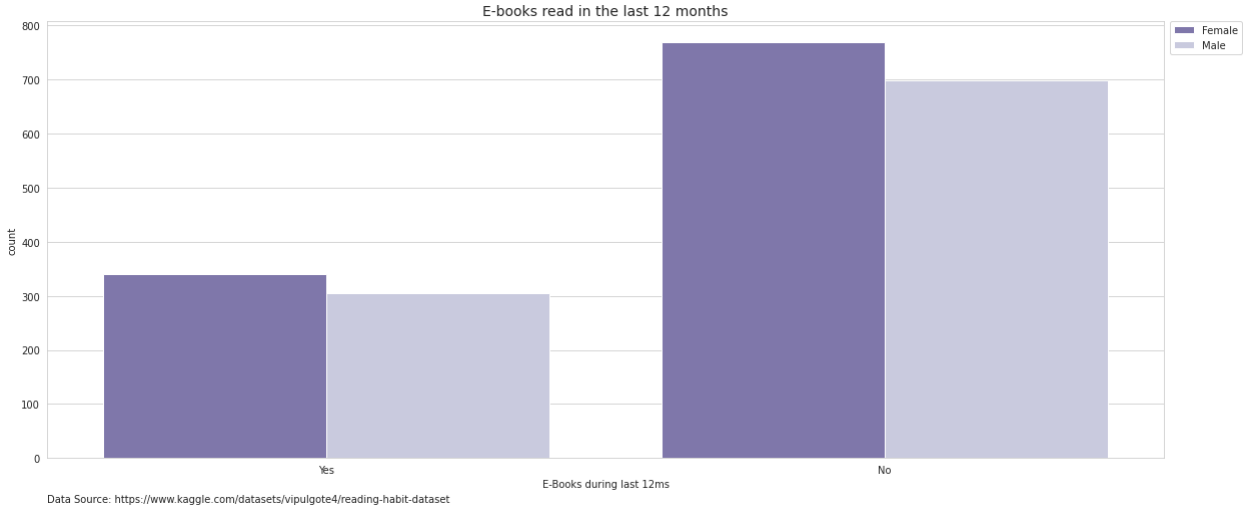



```
plt.figure(figsize=(20,8))
sns.countplot(x = "Audiobooks in last 12ms", data = df, hue = "Sex", order = ["Yes", "No"], hue_order =

plt.title("Audiobooks read in the last 12 months", fontsize = 14)
plt.legend(bbox_to_anchor=(1.005, 1), loc='upper left', borderaxespad=0)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), x
plt.show()
```

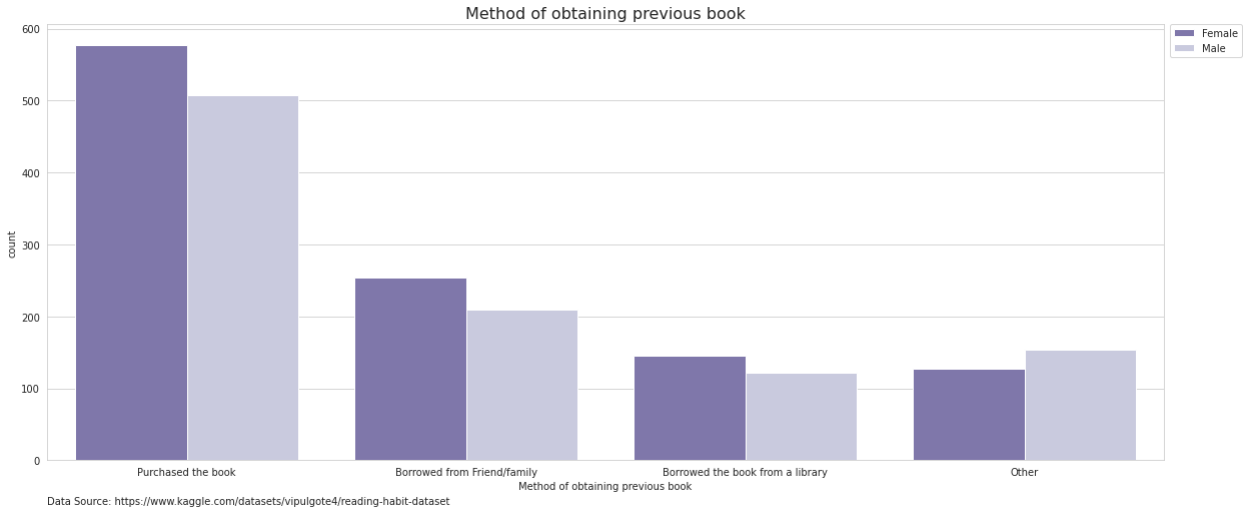



```
plt.figure(figsize=(20,8))
sns.countplot(x = "E-Books during last 12ms",data = df, hue = "Sex", order = ["Yes", "No"], hue_order =

plt.title("E-books read in the last 12 months", fontsize = 14)
plt.legend(bbox_to_anchor=(1.005, 1), loc='upper left', borderaxespad=0)
```

```
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), x
plt.show()
```

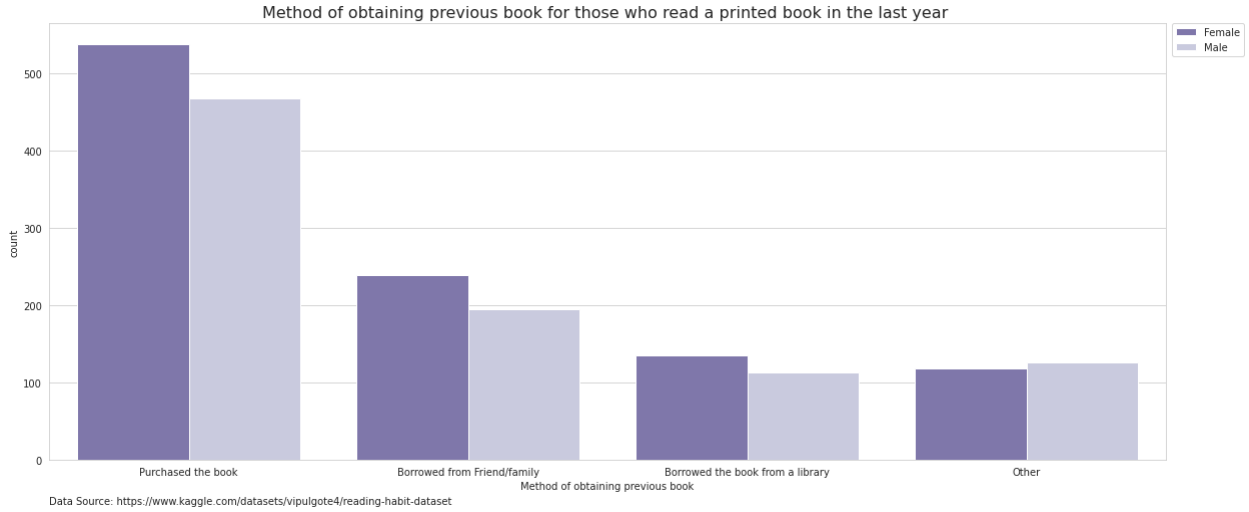



```
plt.figure(figsize =(20,8))
sns.countplot(x = "Method of obtaining previous book", data = df,
              order = ['Purchased the book', 'Borrowed from Friend/family','Borrowed the book from a lib
             hue = "Sex", hue_order = ["Female", "Male"],
              palette = "Blues_r")

plt.title("Method of obtaining previous book", fontsize = 16)
plt.legend(bbox_to_anchor=(1.005, 1), loc='upper left', borderaxespad=0)

plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), x
plt.show()
```

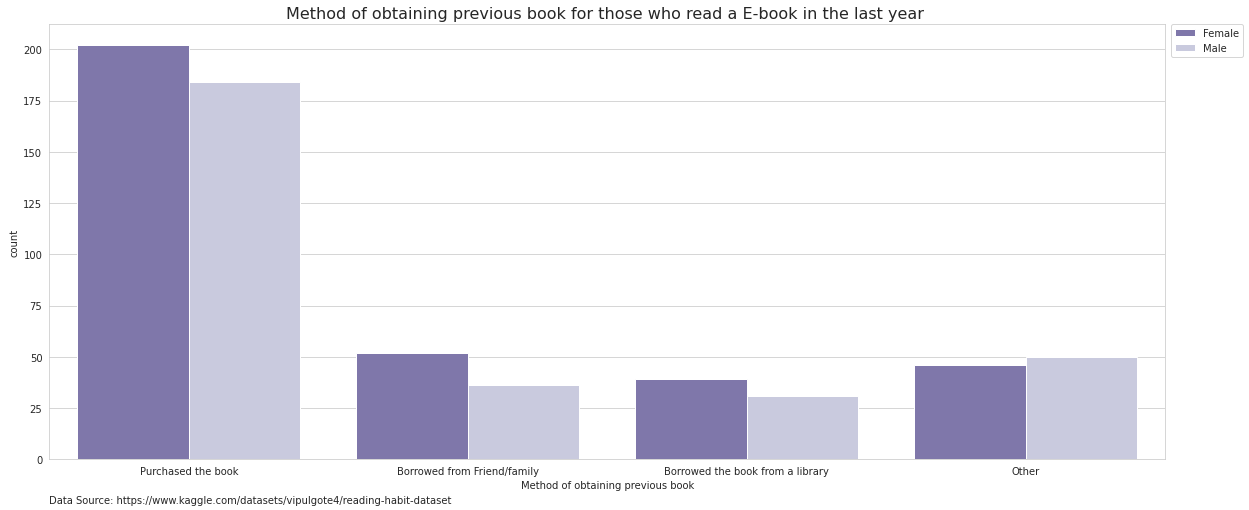



```
plt.figure(figsize =(20,8))
sns.countplot(x = "Method of obtaining previous book", data = df[df["Printed books in the last 12ms"]==
              order = ['Purchased the book', 'Borrowed from Friend/family',
```

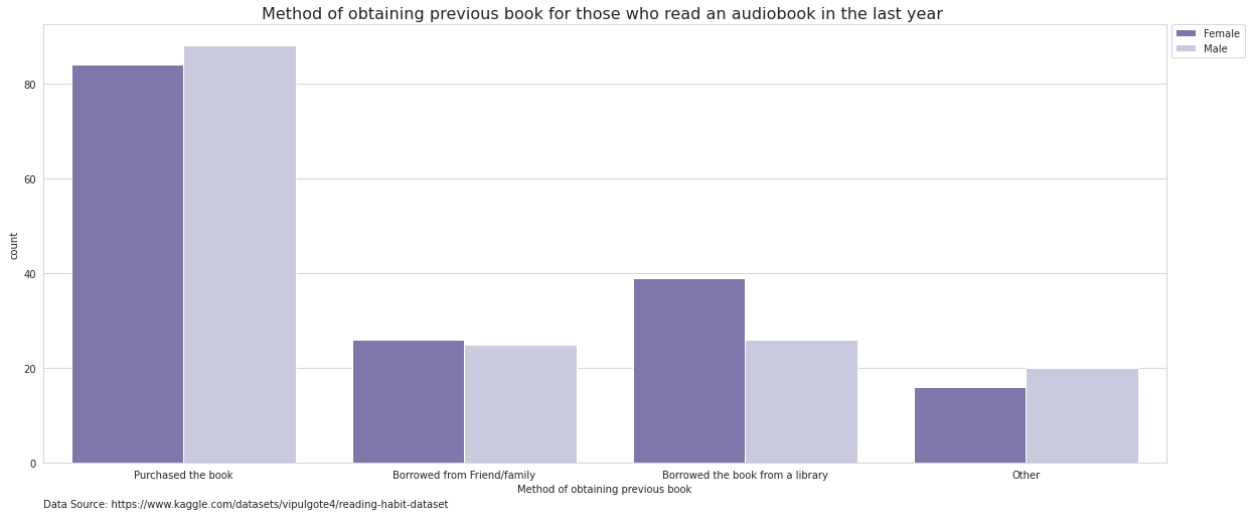
```
       'Borrowed the book from a library', 'Other'],
              hue = "Sex", hue_order = ["Female", "Male"],
              palette = "Purples_r")

plt.title("Method of obtaining previous book for those who read a printed book in the last year ", fonts
plt.legend(bbox_to_anchor=(1.005, 1), loc='upper left', borderaxespad=0)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), x
plt.show()
```




```
plt.figure(figsize =(20,8))
sns.countplot(x = "Method of obtaining previous book", data = df[df["E-Books during last 12ms"]=="Yes"]
              order = ['Purchased the book', 'Borrowed from Friend/family',
       'Borrowed the book from a library', 'Other'],
              hue = "Sex",hue_order = ["Female", "Male"],
              palette = "Purples_r")

plt.title("Method of obtaining previous book for those who read a E-book in the last year ", fontsize =
plt.legend(bbox_to_anchor=(1.005, 1), loc='upper left', borderaxespad=0)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), x
plt.show()
```

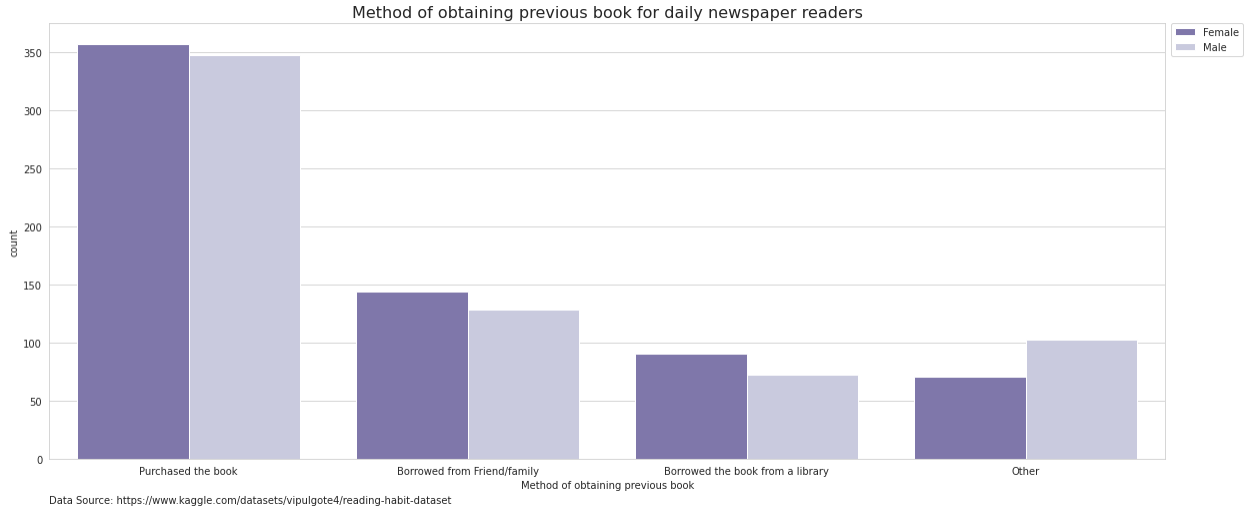



```
plt.figure(figsize =(20,8))
sns.countplot(x = "Method of obtaining previous book", data = df[df["Audiobooks in last 12ms"]=="Yes"],
              order = ['Purchased the book', 'Borrowed from Friend/family',
       'Borrowed the book from a library', 'Other'],
              hue = "Sex", hue_order = ["Female", "Male"],
              palette = "Purples_r")


plt.title("Method of obtaining previous book for those who read an audiobook in the last year ", fontsi
plt.legend(bbox_to_anchor=(1.005, 1), loc='upper left', borderaxespad=0)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), x
plt.show()
```

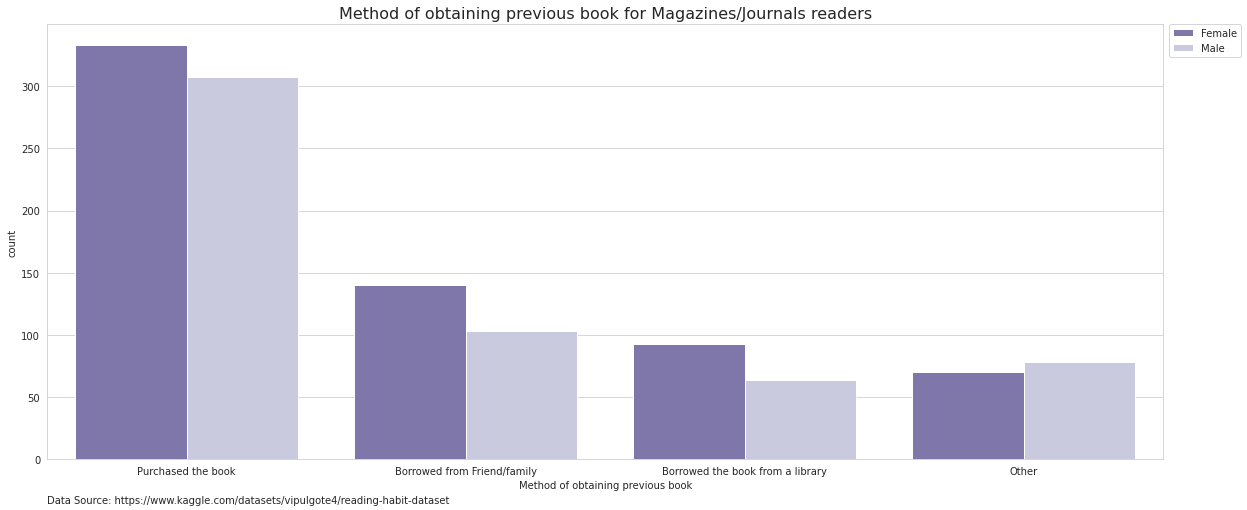



```
# create graph
plt.figure(figsize =(20,8))
sns.countplot(x = "Method of obtaining previous book", data = df[df["Daily news or newspapers"]=="Yes"]
              order = ['Purchased the book', 'Borrowed from Friend/family',
       'Borrowed the book from a library', 'Other'],
              hue = "Sex",
```

```
              palette = "Purples_r")

# format graph
plt.title("Method of obtaining previous book for daily newspaper readers", fontsize = 16)
plt.legend(bbox_to_anchor=(1.005, 1), loc='upper left', borderaxespad=0)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), x
plt.show()
```




```
plt.figure(figsize =(20,8))
sns.countplot(x = "Method of obtaining previous book", data = df[df["Magazines or Journals"]=="Yes"],
              order = ['Purchased the book', 'Borrowed from Friend/family',
       'Borrowed the book from a library', 'Other'],
              hue = "Sex",
              palette = "Purples_r")


plt.title("Method of obtaining previous book for Magazines/Journals readers", fontsize = 16)
plt.legend(bbox_to_anchor=(1.005, 1), loc='upper left', borderaxespad=0)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), x
plt.show()
```




```
plt.figure(figsize=(20,8))
sns.countplot(x = "Daily news or newspapers",data = df, hue = "Sex", palette = "Purples_r" , order = ["Y

plt.title("Daily news or newspapers readers and sex", fontsize = 14)
plt.legend(bbox_to_anchor=(1.005, 1), loc='upper left', borderaxespad=0)
plt.annotate('Data Source: https://www.kaggle.com/datasets/vipulgote4/reading-habit-dataset', (0,-.1), x
plt.show()
```