```
import pandas as pd
import matplotlib.pyplot as plt


# Load the data from the URL
url = "https://www.hubertiming.com/results/2023WyEasterLong"

# Read all tables from the webpage (disable automatic header detection)
tables = pd.read_html(url, header=None)

# --- First Table: Summary Data ---
# The first table contains summary information (e.g., number of finishers)
df_summary = tables[0]

# Rename the columns for clarity
df_summary.columns = ["category", "count"]

# Display the cleaned summary table
print("\n--- Cleaned Summary Table ---")
display(df_summary.head())

# --- Second Table: Race Data for Runners ---
# The second table contains the actual race data (results for each runner)
df_race = tables[1]  # Second table with race data

# Set proper column names for the race data
df_race.columns = ["place", "bib", "name", "gender", "age", "city", "state", "time", "gender_place", "age_group", "age_group_plac

# Optional: Drop any rows where critical columns (like 'name' or 'time') are missing
df_race = df_race.dropna(subset=["name", "time"])

# Convert 'time' column to timedelta (if it's in a string format like "1:33:19")
df_race['time'] = pd.to_timedelta(df_race['time'], errors='coerce')

# Create a new column for 'finish_time_minutes' (in minutes)
df_race['finish_time_minutes'] = df_race['time'].dt.total_seconds() / 60

# Convert 'age' to numeric (some rows might have invalid age data)
df_race['age'] = pd.to_numeric(df_race['age'], errors='coerce')

# Reset index for clarity
df_race = df_race.reset_index(drop=True)

# Display the cleaned race data (first few rows)
print("\n--- Cleaned Race Data Table ---")
display(df_race.head())
```

```
--- Cleaned Summary Table ---
```

	category	count
0	Finishers:	18
1	Male:	11
2	Female:	6
3	Non-Binary:	1

```
--- Cleaned Race Data Table ---
```

	place	bib	name	gender	age	city	state	time	gender_place	age_group	age_group_place	finish_time_minutes
0	1	345	ZACH VIOLETT	M	40	BEND	OR	0 days 01:33:19	1 of 11	M 40-44	1 of 2	93.316667
1	2	335	KYLEE ROOD	O	34	BEND	OR	0 days 01:42:56	1 of 1	O 30-34	1 of 1	102.933333
2	3	323	ZEBEDIAH MILLSLAGL	M	25	BEND	OR	0 days 01:52:07	2 of 11	M 25-29	1 of 2	112.116667
3	4	309	DAVID	M	25	SEATTLE	WA	0 days	3 of 11	M 25-29	1 of 2	113.133333

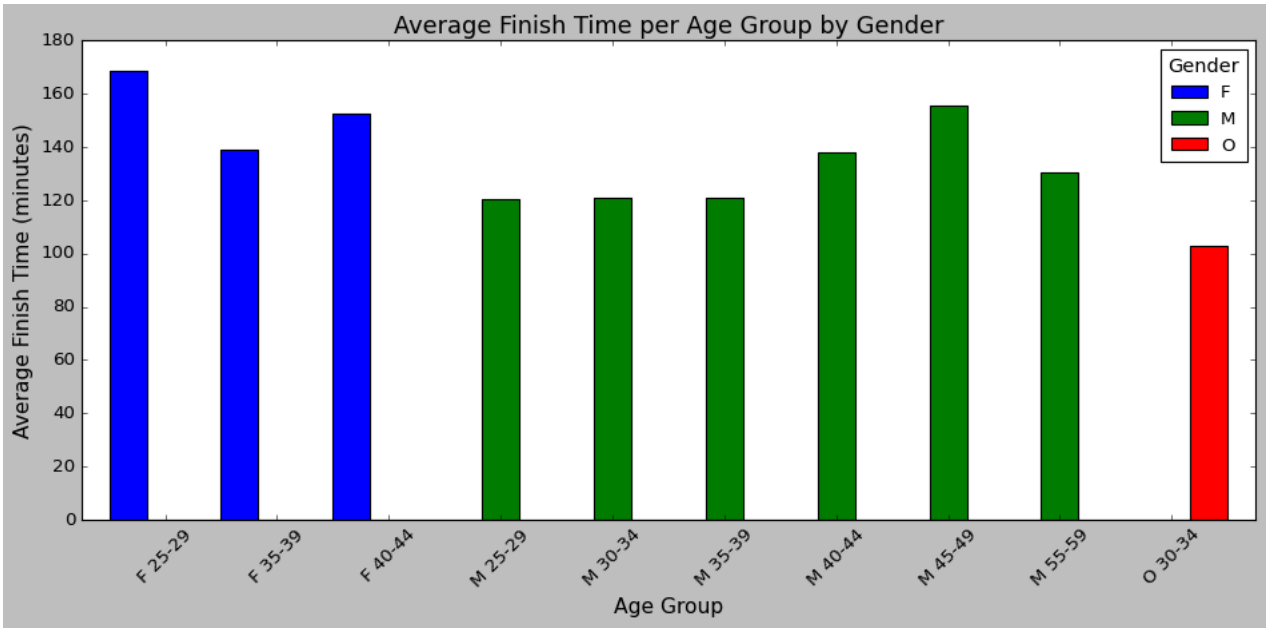
```
# Group by age group and gender, then calculate the average finish time
avg_finish_time = (
    df_race.groupby(["age_group", "gender"])["finish_time_minutes"]
    .mean()
```

```
    .reset_index()
    .sort_values(by=["age_group", "gender"])
)

# Rename column for clarity
avg_finish_time.rename(columns={"finish_time_minutes": "avg_finish_time_minutes"}, inplace=True)

# Display the result
print("\n--- Average Finish Time by Age Group and Gender ---")
display(avg_finish_time)
```

```
--- Average Finish Time by Age Group and Gender ---
```

	age_group	gender	avg_finish_time_minutes
0	F 25-29	F	168.741667
1	F 35-39	F	139.250000
2	F 40-44	F	152.491667
3	M 25-29	M	120.241667
4	M 30-34	M	121.116667
5	M 35-39	M	121.100000
6	M 40-44	M	137.850000
7	M 45-49	M	155.541667
8	M 55-59	M	130.633333
9	O 30-34	O	102.933333

```
import matplotlib.pyplot as plt

# Use classic style (optional)
plt.style.use("classic")

# Pivot the data for grouped bar chart
pivot_table = avg_finish_time.pivot(index="age_group", columns="gender", values="avg_finish_time_minutes")

# Create the plot
ax = pivot_table.plot(
    kind="bar",
    figsize=(12, 6),
    width=1.0,  # Thicker bars that touch
)

# Titles and labels with custom font sizes
plt.title("Average Finish Time per Age Group by Gender", fontsize=16)
plt.xlabel("Age Group", fontsize=14)
plt.ylabel("Average Finish Time (minutes)", fontsize=14)

# Customize axis tick fonts
ax.tick_params(axis='x', labelsize=12)
ax.tick_params(axis='y', labelsize=12)

# Rotate x-axis ticks for readability
plt.xticks(rotation=45)

# Legend
plt.legend(title="Gender", fontsize=12, title_fontsize=13)

# Tight layout for better spacing
plt.tight_layout()

# Show the plot
plt.show()
```


Average Finish Time per Age Group by Gender
Average Finish Time (minutes)
Age Group
Gender
F
M
O
0
20
40
60
80
100
120
140
160
180
F 25-29
F 35-39
F 40-44
M 25-29
M 30-34
M 35-39
M 40-44
M 45-49
M 55-59
O 30-34