To fetch data from a website using an API and add it to an Excel file in real-time using Python, you can follow these steps:

1. Choose a Python library for making API requests (such as **requests** or **http.client**) to fetch the data.
2. Choose a library to manipulate Excel files (such as **openpyxl** or **pandas**).
3. Set up a loop to periodically fetch the data from the API.
4. Parse the JSON response (if the API returns JSON data).
5. Add the fetched data to the Excel file.
6. Save the Excel file.

Here's an example using **requests** for API requests and **openpyxl** for Excel manipulation:

import requests

from openpyxl import Workbook

from datetime import datetime

import time

# Function to fetch data from the API

def fetch\_data\_from\_api():

url = "your\_api\_endpoint\_here"

headers = {

"Authorization": "Bearer your\_api\_token\_here"

}

response = requests.get(url, headers=headers)

if response.status\_code == 200:

return response.json() # Assuming API returns JSON data

else:

print("Failed to fetch data from API")

return None

# Function to add data to Excel

def add\_data\_to\_excel(data, excel\_file):

workbook = Workbook()

sheet = workbook.active

# Assuming data is a dictionary or list of dictionaries

for row in data:

sheet.append(row.values())

# Save the Excel file

timestamp = datetime.now().strftime("%Y-%m-%d\_%H-%M-%S")

excel\_file\_path = f"data\_{timestamp}.xlsx"

workbook.save(excel\_file\_path)

print(f"Data saved to {excel\_file\_path}")

# Main loop to fetch data periodically

def main():

while True:

data = fetch\_data\_from\_api()

if data:

add\_data\_to\_excel(data, "data.xlsx")

time.sleep(60) # Fetch data every 60 seconds

if \_\_name\_\_ == "\_\_main\_\_":

main()

Make sure to replace **"your\_api\_endpoint\_here"** and **"your\_api\_token\_here"** with your actual API endpoint and token. Additionally, adjust the structure of the data and the way it's added to the Excel file based on the actual response format of your API.