```
import requests
from bs4 import BeautifulSoup
import pandas as pd
import time



headers = {
    'User-Agent': 'Mozilla/5.0 (Windows NT 6.3; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/80.0.3987.162 Safari/537.36'
}

base_url = 'https://finance.yahoo.com/markets/mutualfunds/gainers/?start={}&count=25'



columns = ['Symbol', 'Name', 'Price', 'Change', 'Change %', '50 Day Avg', '200 Day Avg', '3 Month Return', 'YTD Return', '52 Wk Change %', '5
all_data = []

max_pages = 340  # limit scraping to 340 pages
empty_page_streak = 0  # counter to stop scraping when multiple pages are empty
max_empty_pages = 5     # stop after 5 consecutive empty pages


for page in range(0, max_pages * 25, 25):
    print(f"Scraping page with start={page}...")
    url = base_url.format(page)

    try:
        response = requests.get(url, headers=headers, timeout=15)
        response.raise_for_status()
    except requests.exceptions.RequestException as e:
        print(f"Failed to fetch page {page // 25 + 1}: {e}")
        continue

    soup = BeautifulSoup(response.text, 'html.parser')
    table = soup.find('table')

    if not table:
        print(f"No table found on page {page // 25 + 1}")
        empty_page_streak += 1
        if empty_page_streak >= max_empty_pages:
            print("Too many empty pages. Stopping scrape.")
            break
        continue
    else:
        empty_page_streak = 0  # reset if we found a table

    rows = table.find_all('tr')[1:]  # skip header

    for row in rows:
        cols = row.find_all('td')
        data = [col.get_text(strip=True) for col in cols]
        if len(data) == len(columns):
            all_data.append(data)

    time.sleep(1)  # be polite
```

```
Scraping page with start=0...
Scraping page with start=25...
Scraping page with start=50...
Scraping page with start=75...
Scraping page with start=100...
Scraping page with start=125...
Scraping page with start=150...
Scraping page with start=175...
Scraping page with start=200...
Scraping page with start=225...
Scraping page with start=250...
Scraping page with start=275...
Scraping page with start=300...
Scraping page with start=325...
Scraping page with start=350...
No table found on page 15
Scraping page with start=375...
No table found on page 16
Scraping page with start=400...
No table found on page 17
Scraping page with start=425...
```

```
No table found on page 18
Scraping page with start=450...
No table found on page 19
Too many empty pages. Stopping scrape.
```

```
# Save results
if all_data:
    df = pd.DataFrame(all_data, columns=columns)
    df.to_csv("yahoo_mutual_fund_gainers.csv", index=False)
    print("Scraping completed. Data saved to yahoo_mutual_fund_gainers.csv")
else:
    print("No data scraped.")
```