

```

# Ben Hallock HW1-DB
# Iterates through urls, grabs date from header, and then grabs data from each card
based on hyperlinks
# Saves data to a csv file as "Year", "Location", "Medal", "Artist", "FigureName"
# Uses requests, BeautifulSoup, and re libraries
# Prints out 125 entries as of oct 14 2024
import re
import csv
import requests
from bs4 import BeautifulSoup

# Define the URLs of the pages to scrape
urls = [
    "https://thegoldendemoncompendium.com/event?id=66151120",
    "https://thegoldendemoncompendium.com/event?id=61723370",
    "https://thegoldendemoncompendium.com/event?id=47607345"
]

# Download robots.txt file
robots_url = "https://thegoldendemoncompendium.com/robots.txt"
robots_response = requests.get(robots_url)
with open('robots.txt', 'w') as robots_file:
    robots_file.write(robots_response.text)

# Prepare a list to store the extracted data
data = []

i = 0

# Iterate through the URLs to scrape data
for url in urls:
    # Send a GET request to the URL
    response = requests.get(url)
    soup = BeautifulSoup(response.content, 'html.parser')

    # Extract the year and location from the h1
    h1_text = soup.find('h1', class_='mb-3').text.strip()
    year_match = re.search(r'\b\d{4}\b', h1_text)
    year = year_match.group(0) if year_match else None
    location = h1_text.replace(year, '').strip() if year else None

    # Find all category headers and the corresponding entries
    categories = soup.find_all('h3', class_='mt-4')

    for category_header in categories:
        category = category_header.text.strip() # Get the category from the header text

        # Get the next sibling divs which contain the entry cards
        card_section = category_header.find_next_sibling('div', class_='row')

        for card in card_section.find_all('div', class_='col-md-4'):
            # Extract medal
            medal = card.find("h3").text.strip() if card.find("h3") else None

            # Extract artist name and figure name
            artist_link = card.find("a", href=re.compile(r"/artist"))
            figure_link = card.find("a", href=re.compile(r"/entry"))

            artist_name = artist_link.text.strip() if artist_link else None
            figure_name = figure_link.text.strip() if figure_link else None

```

```

# Check if all attributes are found
if all([medal, artist_name, figure_name, year, location, category]):
    entry = {
        "Year": year,
        "Location": location,
        "Medal": medal,
        "Artist": artist_name,
        "FigureName": figure_name,
        "Category": category
    }
    data.append(entry)
    print(i)
    print(entry)
    i += 1

# Save the extracted data to a CSV file
with open('golden_demon_winners.csv', mode='w', newline='', encoding='utf-8') as file:
    writer = csv.DictWriter(file, fieldnames=["Year", "Location", "Medal", "Artist", "FigureName", "Category"])
    writer.writeheader()
    writer.writerows(data)

print("Data extraction complete. Saved to golden_demon_winners.csv.")

```