

Copilot

I apologize for the oversight in the previous implementation. Let's correct the functions once again to ensure they pass the provided unit tests:

1. `most_frequent_sites`:

- I've updated the logic to correctly count the number of visits for each site.
- The function now returns the specified number of most frequent sites.
- If the number of sites is greater than the available sites, it returns all available sites.
- If the visits list is empty, it returns an empty set.

```
def most_frequent_sites(visits: list, number: int) -> set:
    """
    Returns a set of most frequent sites visited in total.
    Returns only 'number' of most frequent sites visited.
    :param visits: all visits in browser history
    :param number: number of most frequent sites to return
    :return: set of most frequent sites
    """
    compare_dict = {}
    output_set = set()

    for visit in visits:
        url = visit[0]
        compare_dict.setdefault(url, []).append(visit[4])

    sorted_sites = sorted(compare_dict, key=lambda key: len(compare_dict[key]), reverse=True)

    for i in range(min(number, len(sorted_sites))):
        output_set.add(sorted_sites[i])

    return output_set
```

1. `get_url_info`:

- I've added handling for cases where the visits list is empty or the URL is not found.
- If the visits list is empty, it returns a tuple with empty values.
- If the URL is not found, it returns a tuple with empty values.

```
def get_url_info(visits: list, url: str):
    """
    Returns a tuple with info about the site, which title is passed.
    Function should return:
    title - title of the site with this URL
    last_visit_date - date of the last visit of this site, in format "yyyy-mm-dd"
    last_visit_time - time of the last visit of this site, in format "hh:mm:ss.ms"
    num_of_visits - how many times this site was visited
    average_time - average time spent on this site
    :param visits: all visits in browser history
    :param url: URL of the site to search
    :return: (title, last_visit_date, last_visit_time, num_of_visits, average_time)
    """
    if not visits or url not in [visit[0] for visit in visits]:
        return '', '', '', 0, 0

    filtered_visits = [visit for visit in visits if visit[0] == url]
    num_of_visits = len(filtered_visits)
    total_time = sum(visit[4] for visit in filtered_visits)
    average_time = total_time / num_of_visits if num_of_visits > 0 else 0

    last_visit = max(filtered_visits, key=lambda visit: datetime.strptime(visit[2], '%Y-%m-%d'))
    title, last_visit_date, last_visit_time, _, _ = last_visit

    return title, last_visit_date, last_visit_time, num_of_visits, average_time
```

Feel free to use these corrected functions, and they should now pass the provided unit tests. Let me know if you need any further assistance!