Wikipedia Watching

We will use the Wikipedia Streaming API to generate various reports about the updates being done on Wikipedia websites.

Task 1

Every 1 minute, print the two reports described below to the console. The reports should be based on the data received in the last 1 minute.

Bonus Task

Every 1 minute, print the same reports but now the reports should be based on the data received in the last 5 minute.

Reports

Domains Report

Print the number of the Wikipedia domains that have been updated, followed by a list of the domains sorted by the count of how many unique pages were updated on each. Pages with the same title are assumed to be the same.

Sample Report:

```
Total number of Wikipedia Domains Updated: 10

en.wikipedia.org: 10 pages updated
es.wikipedia.org: 6 pages updated
ru.wikipedia.org: 4 pages updated
hi.wikipedia.org: 1 page updated
...
```

Users Report

Print a list of users that have made changes to **en.wikipedia.org** domain, sorted by their total edit count (available as performer->user_edit_count in each event). If the same user shows up multiple times in the given time period, then use the highest edit count seen for them.

Apart from regular users, various bots also make changes to Wikipedia pages. **For generating this report, any bot users should be excluded.** Whether a user is a bot or not is mentioned in the API response.

Sample Report:

Users who made changes to en.wikipedia.org
James123: 12012
Jiten_Sharma: 8121
...

Wikipedia Event Stream API

Use the data provided by the https://stream.wikimedia.org/v2/stream/revision-create endpoint. This provides a real time feed of all the new revisions being created on Wikipedia.

To get started, you can find Javascript and Python example code and link to libraries here: https://wikitech.wikimedia.org/wiki/Event_Platform/EventStreams

You can see the type of data you will receive from the API here: https://stream.wikimedia.org/v2/ui/#/?streams=revision-create

Notes

- You can complete the task in the programming language of your choice.
- For the bonus task, the report printed at the 1st minute will be based on the data in the first minute, the report at the 2nd minute will be based on data for the first 2 minutes and so on. After 5 minutes, the report printed should be generated based only on the data from the last 5 minutes.

Submission

Setup a local Git repo with your code. The repo should include a README file with instructions on how to setup and run the code. Make a compressed .zip or .tar.gz archive of the repo and send it as an email attachment.