# **Work Summary**

***Methodology:***

**Data Preparation:**

* **R:** Utilized libraries such as jsonlite, tidyverse, urltools, httr, parallel, and doParallel. Data was read from "response.json" with fromJSON(), and nested JSON was flattened using unnest() and mutate() functions.
* **Python:** Used libraries including json, pandas, re, socket, concurrent.futures, and swifter. Data was read from "response.json" with json.load(), and flattened using pandas functions explode() and concat().

**Efficient Data Enhancement:**

* **R:** Extracted domains from URLs with urltools::domain(). Optimized IP fetching by identifying unique domains, and used foreach with doParallel for parallel processing. Created a get\_ip function for DNS queries.
* **Python:** Extracted domains using regex. Implemented optimizations for IP fetching by identifying unique domains and used ThreadPoolExecutor for parallel processing. Created a get\_ip function for efficient DNS querying.

**Data Transformation:**

* **R:** Selected relevant columns, added IP addresses to the dataframe, and saved as flattened\_dmca\_data.csv. Removed columns where all values were null to avoid empty columns.
* **Python:** Selected relevant columns, incorporated IP addresses, and saved as flattened\_response\_domain\_ip.csv. Removed columns where all values were null to avoid empty columns.

**Data Analysis and Summarization (Same in python and R):**

* **Top 10 Domains with Most DMCA Notices:** Grouped by domain, counted notices and URLs, and saved as "top\_10\_infringing\_domains.csv".
* **DMCA Notices Distribution Over Time:** Converted date\_sent to Date format, grouped by date, and saved as "dmca\_notices\_time\_distribution.csv".
* **Top 20 Copyright Holders:** Grouped by principal\_name, calculated notice count and unique domains, saved as " copyright\_holders\_rank\_wise.csv".