Definizione di una funzione che dato l'API di Twitter, la collection delle trending topics e delle available locations, dei tempi di secondi tra ogni request e di reset delle rate limit, prende dall'API di twitter i dati e inserisce senza duplicati le available locations con trends e i rispettivi trending topics nelle due collections.

```python
def getAndAddAvailableLocationsAndTrendingTopics(api_di_twitter,
                                                 collezione_trending_topics,
                                                 collezione_available_locations,
                                                 secondsBetweenEachAppRequestArg,
                                                 secondsToReset):
    dateNow = datetime.today().strftime('%Y-%m-%d')
    dateTimeNow = datetime.today().strftime('%Y-%m-%d-%H:%M:%S')
    woiedsWhereTrendsAvailable = api_di_twitter.trends.available()

    for availableLocation in woiedsWhereTrendsAvailable:

        print("Adding", availableLocation['name'], "to the available locations collection.")

        # Inserimento locations in available locations collection
        locationRecord = {"locations_name": "Worldwide", "locations_woeid": 1, "countries": []}
        locationInDBquery = collezione_available_locations.find({'dateTime': {"$regex": str(dateNow) + '.*'}})

        locationInDB = ""
        for item in locationInDBquery:
            locationInDB = item

        if len(locationInDB) == 0:  # empty locationInDB
            if availableLocation['placeType']['name'] != "Supername":
                if availableLocation['placeType']['name'] == "Country":
                    locationRecord['countries'] = [
                        {"locations_name": availableLocation['name'], "locations_woeid": availableLocation['woeid'],
                         "towns": []}]
                else:  # it's a town
                    locationRecord['countries'] = \
                        [
                            {
                                "locations_name": availableLocation['country'],
                                "locations_woeid": availableLocation['parentid'],
                            }
                        ]
                    towns = {"towns": [{"locations_name": availableLocation['name'],
                                        "locations_woeid": availableLocation['woeid']}]}
                    locationRecord = {**locationRecord['countries'][0], **towns}
            jsonRecord = {"dateTime": dateTimeNow, "locations": locationRecord}
            collezione_available_locations.insert_one(jsonRecord)
```