Backend Take-Home Exercise

Your task is to create an in-memory cache with a REST interface. This cache will store JSON objects (strings) in the server's memory that can be accessed via the API. For simplicity the server is single-threaded. The cache accepts the following configuration parameters:

- **Number of slots (int, default 10,000):** Maximum number of objects to be stored simultaneously in the server's memory. If the server runs out of slots it will behave according to the Eviction Policy setting
- **Time-To-Live (int, default: 3600 secs):** Object's default time-to-live value in seconds if no TTL is specified as part of a write request. If TTL is set to 0 that means store indefinitely (until an explicit DELETE request)
- **Eviction Policy (enum, default: REJECT):** This indicates what to do when the cache runs out of slots. The following options are:
    - OLDEST_FIRST: If there are no slots available the cache will evict the oldest active object and store the new object in its place
    - NEWEST_FIRST: If there are no slots available the cache will evict the newest active object first and store the new object in its place
    - REJECT: When the cache runs out of storage it just reject the store request

The REST API will support the following operations:

- **`GET /object/{key}`**
    - This will return the object stored at {key} if the object is not expired.
    - Returns
        - 200: If the object is found and not-expired
        - 404: If the object is not found or expired
- **`POST or PUT /object/{key}?ttl={ttl}`**
    - This will insert the {object} provided in the body of the request into a slot in memory at {key}. If {ttl} is not specified it will use server's default TTL from the config, if ttl=0 it means store indefinitely
    - Returns
        - 200: If the server was able to store the object
        - 507: If the server has no storage
- **`DELETE /object/{key}`**
    - This will delete the object stored at slot {key}
    - Returns
        - 200: If the object at {key} was found and removed
        - 404: If the object at {key} was not found or expired

Hints

- You can use your programming language of choice
- Build your solution as you'd build production code