

Fundamentos de Informática - Final Práctico

Fecha: 07/07/2022

Ejercicio:

La compañía **Capsule Corp.** fundado por el Dr. Brief, se encuentra desarrollando un nuevo producto que podría generar otra revolución para los ciudadanos de la Capital del Oeste como para el resto del mundo.

Sabemos que la invención y la producción de las *Cápsulas Hoi Poi* es extremadamente popular y útil, la Corporación Cápsula es una de las empresas más grandes, ricas e influyentes en el mundo. Aunque el producto primario de la corporación Cápsula es la tecnología de las cápsulas Hoi Poi, la compañía también desarrolla y produce muchos de los elementos encontrados dentro de cápsulas.



Con el lanzamiento de este nuevo producto la compañía apunta a conquistar un nuevo mercado, que todavía se encuentra virgen.

Hemos sido contratados por el equipo encargado del proyecto, el mismo denominado "**Prometheus 009**", luego de un pesado proceso de selección nos dan la bienvenida y nos comentan cuál sería nuestra tarea a realizar.

El producto se trata de un dispositivo en forma de brazalete capaz de leer nuestro ADN, flujo sanguíneo y emociones corporales, con el fin de acumular la data y darnos un estado promedio del día.

Para cumplir con la primera tarea de prototipado, debemos desarrollar una API, la cual utilizará el dispositivo, y debe cumplir con las siguientes características:

- El *base_URL* de la API será: "<host>:<port>/api/prometheus"
- Contar con un endpoint tipo **POST** con la siguiente URi `"/statistics"` que permita recibir un body request con la estructura que figura al final, y persistir la data en un archivo `"statistics.csv"`. No olvidar de crear el header.
- Contar con otro endpoint tipo **GET** con la siguiente URi `"/statistics"`, que devuelva todos los datos obtenidos del archivo anterior.
- Finalmente nos piden que desarrollemos un endpoint tipo **GET** con la siguiente URi `"/statistics/reports"`, el cual será mandatorio que se envíe un parámetro `"report_date"` con la fecha en este formato `"YYYY-MM-DD"`, ejemplo: `"2022-07-01"`. En caso que el parámetro no venga definido, devolver un código



HTTP 400, indicando que el parámetro es obligatorio. Definir la estructura de error que desee. En caso satisfactorio, es decir que el parámetro esté definido con el formato nombrado, deberemos devolver para la fecha consultada, el promedio de "blod_sugar_level" y "emotion_level". Definir un **JSON** response con 2 propiedades:

- avg_blod_sugar_level
- avg_emotion_level

Body request (estructura)

```
{
  "date": "2022-06-01",
  "dna": "CCGCGCATCCTATTTATTAAGTTAATTCTATAG"
  "blod_sugar_level": 138,
  "emotion_level": 19
}
```

*El éxito del producto depende en parte de esta API
Buena suerte!!*

Hints:

- Utilizar Flask para desarrollar la API
- Utilizar Postman como cliente de su API
- Estructurar el proyecto correctamente y se valora el uso de comentarios
- Utilizar POO y definir en el/los objetos el método "serialize(self)" para ayudarnos en la serialización del objeto.
- Para calcular el promedio se recomienda utilizar el módulo "import statistics". Ej. `x = statistics.mean([1, 2, 3, ...])`

Condiciones de entrega:

Todos los archivos del ejercicio deberán ser comprimidos utilizando *zip, rar, 7zip u otra*. El nombre del archivo deberá ser tal cual como se indica, y el mismo subido al Campus:

Generico: "nombre_apellido_fdi_final_07_07_2022.zip"

Alumno ej: "josefina_villar_fdi_final_07_07_2022.zip"