

Junior

Escenario Re-Id

Hay algunos disturbios en las filas de los súbditos: los súbditos con números de identificación como "1", "42" y otros números "buenos" se han enseñoreado de los pobres súbditos que tienen identificaciones más aburridas.

Para sofocar los disturbios, el comandante Lambda te ha asignado la tarea de reasignar a todos nuevos ID aleatorios basados en un esquema completamente infalible. Commander Lambda ha concatenado los números primos en una sola cadena larga: "2357111317192329...".

Ahora cada minion debe sacar un número de un sombrero. Ese número es el índice inicial en esa cadena de números primos, y el nuevo número de identificación del minion serán los siguientes cinco dígitos de la cadena. Entonces, si un minion saca "3", su número de identificación será "71113".

Ayude al comandante a asignar estos ID escribiendo una solución de función (n) que toma el índice inicial n de la cadena de Lambda de todos los números primos y devuelve los siguientes cinco dígitos en la cadena. Commander Lambda tiene muchos secuaces, por lo que el valor de n siempre estará entre 0 y 10000.

Restricciones:

Lenguaje	FrameWork	Arquitectura
JavaScript NodeJs	Express	CLEAN

Los números de cada minion deben ser consultados mediante una solicitud REST con un método GET

```
${baseUrl}/minion/${codeIdToAsk}
```

la respuesta de esta solicitud debe ser un JSON

```
{
  "minionAsk" : "${codeIdAsked}",
  "relatedCode": "${PrimeCode}"
}
```

cada registro consultado deberá ser guardado en una base de datos PostgreSQL

Casos de ejemplo

Input : 0

```
${baseUrl}/minion/0
```

Response

```
{
  "minionAsk" : "0",
  "relatedCode": "23571"
}
```

Input : 3

```
${baseUrl}/minion/3
```

Response

```
{
  "minionAsk" : "0",
  "relatedCode": "71113"
}
```

Entregables

Código	Base de datos	Documentación
Repositorio Github/Gitlab	Diseño y creación	Diagramas de secuencia, casos de uso