

## Challenge técnico: Career Switch 2022

¡Bienvenidos al challenge técnico de Rooftop Academy! Estamos contentos de que nos elijas como motor de crecimiento para tu futuro. En este documento, encontrarás toda la información relacionada a este desafío (los ejercicios en sí, formato de entrega, etc). Por eso, **te recomendamos que lo leas detenidamente antes de comenzar.**

Para aprobar el ingreso, deberás solucionar la consigna propuesta.

## Changelog

Acá se agregarán todos los cambios al documento desde su difusión:

- 2021-06-20: Se ha cambiado la fecha de entrega para el 28 de junio (debido a un error de escritura, se había puesto incorrectamente "Julio", aunque en realidad es "Junio").
- 2021-06-20: [Solucionado] La API ha tenido problemas durante un lapso en el día de hoy. **Actualmente está funcionando correctamente.** Pedimos disculpas por las molestias.
- 2021-06-22: La API ahora tiene una cantidad máxima de requests por minuto, para evitar que algoritmos ineficientes desborden de peticiones. Si en algún momento recibes como respuesta un código HTTP 429, significa que has alcanzado el límite. Espera unos minutos, y podrás realizar nuevamente requests ([Ver límite](#)).

## Entrega

Para entregar la solución tendrás que crear un repositorio en tu cuenta de Github, o del servicio Git que utilices. **Asegúrate que el repositorio sea público para que podamos verlo.**

Cuando termines de resolver envíanos un correo a [academy@rooftop.dev](mailto:academy@rooftop.dev). **Como asunto (o subject) del email deberás poner “Challenge ingreso Career Switch”** (sin comillas, y respetando las mayúsculas, minúsculas y espacio).

**En el cuerpo del email deberás ingresar tu nombre y apellido y la URL del repositorio.**

Solamente aceptaremos links a repositorios, no es necesario que envíes links de Google Drive, ni que adjuntes archivos comprimidos tipo Zip en el mensaje.

**En el caso de que se reciba correctamente tu solución** (con un asunto válido), **se responderá un email automático que te confirmará que hemos recibido correctamente la entrega.**

**Tendrás tiempo de presentar la solución hasta el 28 de junio de 2022 a las 08:00** (hora Argentina - GMT - 3). **Posterior a la misma no se aceptarán más entregas.**

**Sólo se tomará en cuenta los commits realizados hasta la fecha de cierre.** Si agregas cambios en el repositorio luego, no será tomado en cuenta para la validación. Solo evaluaremos la rama master o main.

## CONDICIONES:

- Resolver la consigna en alguno de los siguientes lenguajes: **Java, Javascript, Typescript, PHP, Go, Python, Ruby, C#.**
- Si es necesario, incluir en el README del proyecto los pasos a seguir para que podamos levantar/ejecutar el proyecto.
- La honestidad es un valor fundamental para nosotros. **Si detectamos soluciones iguales, todas las personas implicadas serán bloqueadas de Rooftop Academy y no podrán ingresar a ningún programa ahora ni en el futuro.** No intentes copiar ni prestar la solución a alguien, porque ambos resultarán perjudicados.
- Si no se respeta el formato de entrega (asunto correcto, cuerpo de email con nombre y apellido y una url) se descontarán puntos.

## Tengo una duda / Encontré un error en el documento

Si tienes alguna duda, o encuentras algún error en este documento, por favor contáctanos a [soporte@rooftopacademy.com](mailto:soporte@rooftopacademy.com). Te responderemos a la brevedad.

## Consigna

El objetivo de este challenge es desarrollar una función que resuelva un acertijo. Una API te devolverá un array de strings desordenados, y deberás ordenarlo con la ayuda de un endpoint, que te guiará en el proceso. Además, **deberás agregar al menos un test** (unitario o funcional) a tu proyecto.

En <https://rooftop-career-switch.herokuapp.com> podrás encontrar información de la API. ¡Pero vamos a explicarlo detalladamente aquí con un ejemplo!

## Obtención de token

El primer paso es realizar una petición GET a <https://rooftop-career-switch.herokuapp.com/token?email=usuario@gmail.com> (reemplazar por tu email), para obtener un API Token. Ese dato es únicamente tuyo, no lo compartas, no lo pierdas y no lo publiques en el repositorio.

Supongamos que esa petición nos da la siguiente respuesta:

```
{
  "token": "R5cCI6Ik-iwib-F0Ij-IyfQ-iJIUzI1NiIsI"
}
```

## Obtener los bloques

Posteriormente, realiza una petición GET a <https://rooftop-career-switch.herokuapp.com/blocks?token=aqui-agregas-el-token> (siguiendo nuestro ejemplo, sería <https://rooftop-career-switch.herokuapp.com/blocks?token=R5cCI6Ik-iwib-F0Ij-IyfQ-iJIUzI1NiIsI>).

Imaginemos que eso nos devuelve lo siguiente:

```
{
  "data": ["qwer", "asdf", "zcvf", "erty", "jhgf", "polk", "gthu", "uhgt"],
  "chunkSize": 4,
  "length": 32
}
```

Dentro de la propiedad “data” se encuentra el array de strings desordenados, que deberás ordenarlo con la ayuda de otro endpoint.

**Muy Importante:** Siempre el dato de la 1era posición del array se encuentra en su ubicación correcta. En el ejemplo el string “qwer” te servirá como punto de partida para comenzar a buscar cual string del resto de la lista es el que le sucede. Para los demás strings a partir de la posición 1 asumimos que podrían estar en posiciones equivocadas.

## Entregable

El repositorio deberá contener al menos dos cosas: **un archivo con una función “check”, que resuelve el acertijo, y al menos un test que pruebe dicho método.**

## Función “check”

El primer objetivo del challenge es que desarrolles una función “check” que reciba como parámetros:

1. Un array de strings (que representa los bloques desordenados).
2. Un token (el cual será usado para llamar a la API).

Asimismo, esta función deberá devolver otro array con los elementos del array “blocks” ordenados de manera secuencial. La firma sería la siguiente (suponiendo que lo programase con Typescript):

```
function check(blocks: Array<string>, token: string): Array<string>
{
  //Desarrollar aquí dentro el algoritmo que ordene los bloques, usando
  // la API "/check".
  //IMPORTANTE: observar que está recibiendo un parámetro "token". El
  // mismo es para usarlo en la llamada a la API.
  return arrayOrdenado;
}
```

¿Cómo saber el orden correcto de los bloques? Existe un endpoint de la API el cual te dirá si dos bloques son consecutivos. Si haces una llamada POST a <https://rooftop-career-switch.herokuapp.com/check?token=aqui-agregas-el-token> (siguiendo nuestro ejemplo, sería a <https://rooftop-career-switch.herokuapp.com/check?token=R5cCl6lk-iwib-F0lj-lyfQ-iJIUzl1NilsI>) , le puedes pasar en el cuerpo un JSON con los bloques que se quiere validar. Supongamos que queremos que validar que "qwer" y "jhgf" están en secuencia:

```
{
  "blocks": [
    "qwer",
    "jhgf"
  ]
}
```

Si nos devuelve como respuesta `{"message": true}`, significa que "qwer" y "jhgf" son secuenciales (es decir, que "jhgf" es el elemento que le sigue a "qwer"). Si devuelve `{"message": false}`, nos indica que no son contiguos.

Finalmente si crees que lograste ordenar el string correctamente, puedes verificarlo en el mismo endpoint, en la URL <https://rooftop-career-switch.herokuapp.com/check?token=aqui-agregas-el-token>.

El body de la request debe contener una clave llamada “encoded” que sea del tipo string y que tenga solamente el string completo de todos los fragmentos ordenados. Ejemplo:

```
{  
  "encoded": "qwerzcvfuhgtasdfertyjhfgthupolk"  
}
```

El cual, si es correcto, devolverá:

```
{  
  "message": true  
}
```

## Tests

**Deberás agregar a tu proyecto al menos un test, que pruebe que la función “check” es correcta.** Es importante que este test **use un mock** para evitar una llamada a la API.

Podrás elegir la librería de testing que más prefieras: JUnit, Rspec, Jest, PHPUnit, etc. Recuerda agregar dicha dependencia al gestor de paquetes que uses. Por ejemplo, al package.json (en caso de usar JavaScript con Node.js), composer.json (en caso de usar PHP), etc. Indicar en el README.md los comandos a ejecutar para instalar dicha dependencia.

## Consideraciones a tener en cuenta

- Debes intentar hacerlo de la manera más óptima, con la menor cantidad de solicitudes al servidor.
- Cuidado al usar funciones o estructuras iterativas y/o recursivas, asegúrate que no produzcan bucles infinitos. Testearlo localmente antes de intentarlo con la API real.

## ¿Cómo pruebo que mi código sea correcto?

En el repositorio <https://github.com/RooftopAcademy/career-switch-challenge-test> encontrarás un programa en cada lenguaje, en donde podrás reemplazar el contenido de la función “check” con lo que has hecho. De esta forma, comprobarás si el código entregado cumple con las características (puedes pensarlo como un test automático ad-hoc).

## Endpoints

Para más información de la API, consultar la URL <https://rooftop-career-switch.herokuapp.com/>.

### GET /token

Requiere un email como parámetro en el query string.  
Devuelve un token que usarás en las consultas a la API.

### GET /blocks

Requiere el token como parámetro en el query string.  
Devuelve un array de fragmentos (“bloques”) desordenados.

### **POST /check**

Requiere el token como parámetro en el query string.  
Devuelve solamente un boolean como respuesta. Si encontraste una combinación válida devolverá “true”, o de lo contrario “false”.

## **Límites de consultas a la API**

La API permite una cantidad máxima de “requests” por minuto por IP, para evitar que algoritmos ineficientes desborden de peticiones. Si en algún momento recibes como respuesta un código HTTP 429, significa que has alcanzado el límite. Espera unos minutos, y podrás realizar “requests” nuevamente.

Cantidad máxima de requests permitida por IP: 150 requests cada 2 minutos.

---

## **Preguntas Frecuentes** 🤔

P: *¿Cuándo sabré si he aprobado esta etapa?*

R: Desde el 29 y 30 de junio nos estaremos comunicando con cada persona que haya enviado el challenge, para comunicarle el resultado.

P: *¿Debo usar alguna base de datos?*

R: No, no necesitarás de ninguna base de datos para resolver este challenge.

P: *¿La calidad del código importa?*

R: Sí, y es tan importante como que funcione y pase los tests. Escribe el código lo mejor que puedas, ¡demuestra lo mejor que sabes hacer!

P: *No llegué a terminar algo, ¿puedo presentarlo igual?*

R: ¡Por supuesto! Si aprueba al menos el 80% de los tests vamos a tenerlo en cuenta.

P: *¿Puedo usar algún framework?*

R: Consideramos que no necesitas un framework para resolver esto, puedes hacerlo directamente con lo que te provee el lenguaje. Pero es aceptable si utilizas alguna librería, como la usada para realizar el test (ya sea JUnit, PHPUnit, etc).

P: *¿Puedo añadir alguna otra cosa a mi repositorio, como un video o gif demostrando como funciona?*

R: ¡Sí! Si es un video, súbelo a un sitio externo (como Drive) debido a que normalmente ocupan mucho espacio.

P: ¿Puedo probar con más bloques?

R: ¡Sí! Si usas otro email en la obtención del token, tendrás otros bloques completamente diferentes para que puedas evaluar.

## Ayuda

¿Encontraste algún error? Por favor, envíanos un email a [soporte@rooftopacademy.com](mailto:soporte@rooftopacademy.com).

## Consejos y recomendaciones

- **La prioridad es que funcione.** El tiempo para resolver la consigna es suficiente, pero puede ser escaso si tienes otras ocupaciones. En primer lugar intenta resolverlo como sea, aunque sientas que estás haciendo las cosas “mal”. Al finalizar, si te sobra tiempo y sabes cómo mejorar el código puedes hacerlo.
- **Estima tu tiempo.** Antes de comenzar a resolver, procura determinar cuanto tiempo maximo le puedes dedicar a cada parte de la tarea. Si alguna dificultad te lleva más tiempo de lo que estimaste, puedes consultarnos para darte alguna orientación. Evita llegar al ultimo día y justificarte diciendo “me tardé mucho al principio intentando hacer...”
- **No busques atajos.** Esta consigna es propia de Rooftop Academy, no la extrajimos de ningún sitio de desafíos de código. Aprovecha el tiempo en mostrar tu experiencia, no caigas en la trampa de buscar la solución resuelta en algún lugar, porque lo más probable es que no la encuentres y te quedes con menos tiempo.
- **Escribe tu progreso.** Si pasas a la siguiente instancia de entrevista personal es muy probable que te preguntemos acerca de lo que hiciste, así como de los pasos que fuiste realizando. Si lo consideras útil, anota día a día tu progreso, los mayores desafíos, los errores y cómo los resolviste. ¡Prepárate para contarnos la experiencia!

Esto es todo lo que necesitas para resolver este ejercicio. ¡Buena suerte!