



**Developer Test**

Con el paso de los años la biblioteca **albo** de comics, necesita tener actualizado todo el listado de escritores, editores y coloristas de cómics que han estado involucrados en las historias de los siguientes integrantes de los Vengadores (Iron Man, Captain America). Así como todos los demás héroes que a través de cada cómic han interactuado con cada uno de ellos. Esto hay que actualizarlo a diario, ya que hay que pagarles regalías respectivas a los escritores, editores y coloristas.

Para esto escribiremos dos servicios:

## Servicio a)

Obtendremos a los editores, escritores y coloristas que han estado involucrados en los cómics del personaje.

Entrada (Alguna de las siguientes URL's)

<http://test.albo.mx/marvel/colaborators/ironman>

<http://test.albo.mx/marvel/colaborators/capamerica>

Salida (De acuerdo al héroe los datos pueden cambiar)

```
{
  last_sync: "Fecha de la última sincronización en dd/mm/yyyy
hh:mm:ss",
  editors : [
    "Wilson Moss","Andy Smidth", ...
  ],
  writers : [
    "Ed Brubaker","Ryan North", ...
  ],
  colorists : [
    "Rico Renzi", ...
  ]
}
```

## Servicio b)

Obtendremos a los otros héroes con los cuales nuestro personaje ha interactuado en cada uno de los cómics.

Entrada (Alguna de las siguientes URL's)

<http://test.albo.mx/marvel/characters/ironman>  
<http://test.albo.mx/marvel/characters/capamerica>

Salida (De acuerdo al héroe los datos pueden cambiar)

```
{
  last_sync: "Fecha de la última sincronización en dd/mm/yyyy
hh:mm:ss",
  characters : [
    {
      character: "Squirrel Girl",
      Comics :[
        "The Unbeateable Squirrel Girl (2015) #38",
        "The Unbeateable Squirrel Girl (2015) #39"
      ]
    },
    {
      character: "Jocasta",
      Comics :[
        "Tony Stark: Iron Man (2018) #2",
        "Tony Stark: Iron Man (2018) #3", ....
      ]
    }, ...
  ]
}
```

## Consideraciones:

- Hay que usar el API de Marvel Studios disponible en <https://developer.marvel.com>
- Hay que registrarse y obtener su **api-key** correspondiente para poder acceder al API.
- El código debe de estar alojado en **github.com**, y ser disponible publicamente.
- El proyecto debe de construirse usando **maven** o **gradle**, la instancia donde se llevará a cabo la evaluación solamente cuenta con estos dos builders.
- Se debe desarrollar con el lenguaje de programación **NodeJS**
- Se deben incluir pruebas unitarias

El proyecto debe de incluir un par de scripts llamados:

**assemble.sh** el cual debe de hacer la construcción del aplicativo que exponen los servicios a) y b).

**avengers.sh** el cual ejecuta el aplicativo que exponen los servicios a) y b), en el puerto 80

El almacenamiento utilizado para la sincronización es libre: MySQL, Postgress, MongoDB, Cassandra, HBase.. etc.

### Criterios de Evaluación

- Ejecución
- Pulcritud de código
- Modularidad
- Validaciones
- Excepciones
- Manejo de errores
- Simplicidad del Código
- Documentación
- Performance
- Tipo de entrega