

## TECHNICAL TEST BE

### V2.0.0

November, 2021

## PRERREQUISITOS

- Tener Git instalado
- Tener Node.js Instalado (v 12.18.1 o superior)
- Clonar el repositorio con la url que has recibido vía email
- Desde la consola, Ve a la carpeta src y ejecuta el comando: `npm install`
- Desde la consola, ejecuta el comando: `npm run test:watch`, and happy code! :)

## RESTFUL FEATURE

PENDING\_PENDING\_PENDING

- Obtener los 3 libros más prestados en el segundo trimestre del año 2021

Necesitamos que construyas una función en `src/top3MostLentBooksQ22021.js` llamada “top3MostLentBooksQ22021”, esta función debe hacer un request de tipo GET a esta url: <https://hiring.condorlabs.io/api/books/logs> con el fin de obtener la data necesaria para calcular y devolver un arreglo con los 3 libros más prestados en el segundo semestre (abril-junio) del año 2021

- Obtener los 5 libros menos prestados en el primer trimestre del año 2021

Necesitamos que construyas una función en `src/top5LessLentBooksQ12021.js` llamada “top5LessLentBooksQ12021”, esta función debe hacer un request de tipo GET a esta url: <https://hiring.condorlabs.io/api/books/logs> con el fin de obtener la data necesaria para calcular y devolver un arreglo con los 5 libros menos prestados en el primer semestre (enero-marzo) del año 2021

- Obtener el libro más prestado por trimestre del año 2021

Necesitamos que construyas una función en `src/mostLentBookForQuarter2021.js` llamada “mostLentBookForQuarter2021”, esta función debe hacer un request de tipo GET a esta url: <https://hiring.condorlabs.io/api/books/logs> con el fin de obtener la data necesaria para calcular y devolver un arreglo con el libro más prestado por trimestre del año 2021

## UNIT TESTING

En el repositorio puedes encontrar unos unit test precontruidos que al inicio del test están en rojo, tu misión es ponerlos todos en verde agregando el código que satisface los requerimientos, tú NO debes modificarlos ya que están creados con el objetivo de evaluar las funcionalidades requeridas. Abre una consola y ejecuta el comando debajo para mantener los test corriendo mientras escribes el código. `npm run test:watch` O ejecutarlos cuando hayas finalizado la prueba. `npm run test`

Abre una consola y ejecuta el comando debajo para mantener los test corriendo mientras escribes el código.

```
npm run test:watch
```

O ejecutarlos cuando hayas finalizado la prueba.

```
npm run test
```

## EXAMPLE:

En este caso, puedes ver un ejemplo de cómo jest muestra la interfaz mientras corre los tests.

```
PASS src/components/__tests__/CustomTaskListComponent.spec.jsx
CustomTaskListComponent
  ✓ should render demo component (16ms)

Test Suites: 1 passed, 1 total
Tests:       1 passed, 1 total
Snapshots:   0 total
Time:        2.027s
Ran all test suites.

Watch Usage
  > Press f to run only failed tests.
  > Press o to only run tests related to changed files.
  > Press q to quit watch mode.
  > Press p to filter by a filename regex pattern.
  > Press t to filter by a test name regex pattern.
  > Press Enter to trigger a test run.

shagman@shagman-next:/mnt/DAS_MAIN/PRFT/plugin-new$ npm run test
```