**API REST DESAFIO**

**Datos a tener en cuenta:**

**Nombre y Apellido: Yanquetruz Gabriel**

**DNI: 44.530.561**

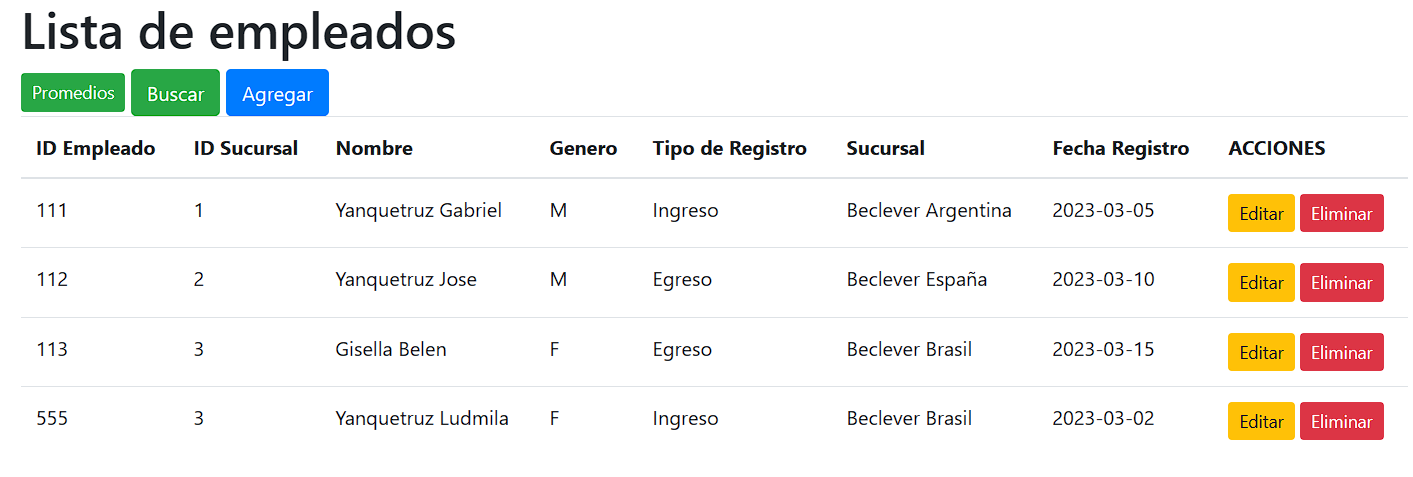
**Formatos de Fecha A respetar:** “YYYY-MM-DD”

El URL varia según donde ejecuta el proyecto, en mi caso es LocalHost:8080 y use MySQL WorkBench.

1. El proyecto está realizado en Spring Tool Suite 4 es una carpeta llamada api-rest-desafío simplemente es descargarla y colocarla en su debido funcionamiento, ejecutandolo ya sea de modo local.
2. Descargar la base de datos llamada Beclever DB o bien colocar el código SQL para crear la base de datos con sus tablas correspondientes.
3. Información Importante: El proyecto se puede probar de dos maneras, de manera FrontEnd y BackEnd

Utilizan la barra de /api/ en el navegador en caso de que sean BackEnd

1. Si deseas utilizar el FrontEnd basta con ingresar a la dirección local, en mi caso es localhost:8080 y automáticamente se mostrará una tabla con todos los datos que se registran en la base de datos. Un ejemplo visual:



Se mantienen las opciones básicas de un CRUD, ya sea Promedios, Buscar, Agregar, Editar y Eliminar empleados.

1. Si deseas utilizar el BackEnd, tendremos disponibles los siguientes comandos solicitados:
2. para poder ejecutar este tipo de comando, deberemos hacer un Post, ya sea con Postman y rellenar su body con dichas variables, la url en mi caso es localhost:8080/api/register

recomiendo preferentemente cambiar el businessLocation por branch id: un ejemplo para un Body

{

"idEmployee": Id De Empleado,

"registerType": "Tipo de Registro",

"descriptionFilter": "Nombre o Apellido",

"gender": "M",

"entryCount": 0,

"exitCount": 0,

"date": "Fecha de Ingreso",

"businessLocation": null,

"branch": {

"id": 1,

"nombre": "Beclever Argentina"

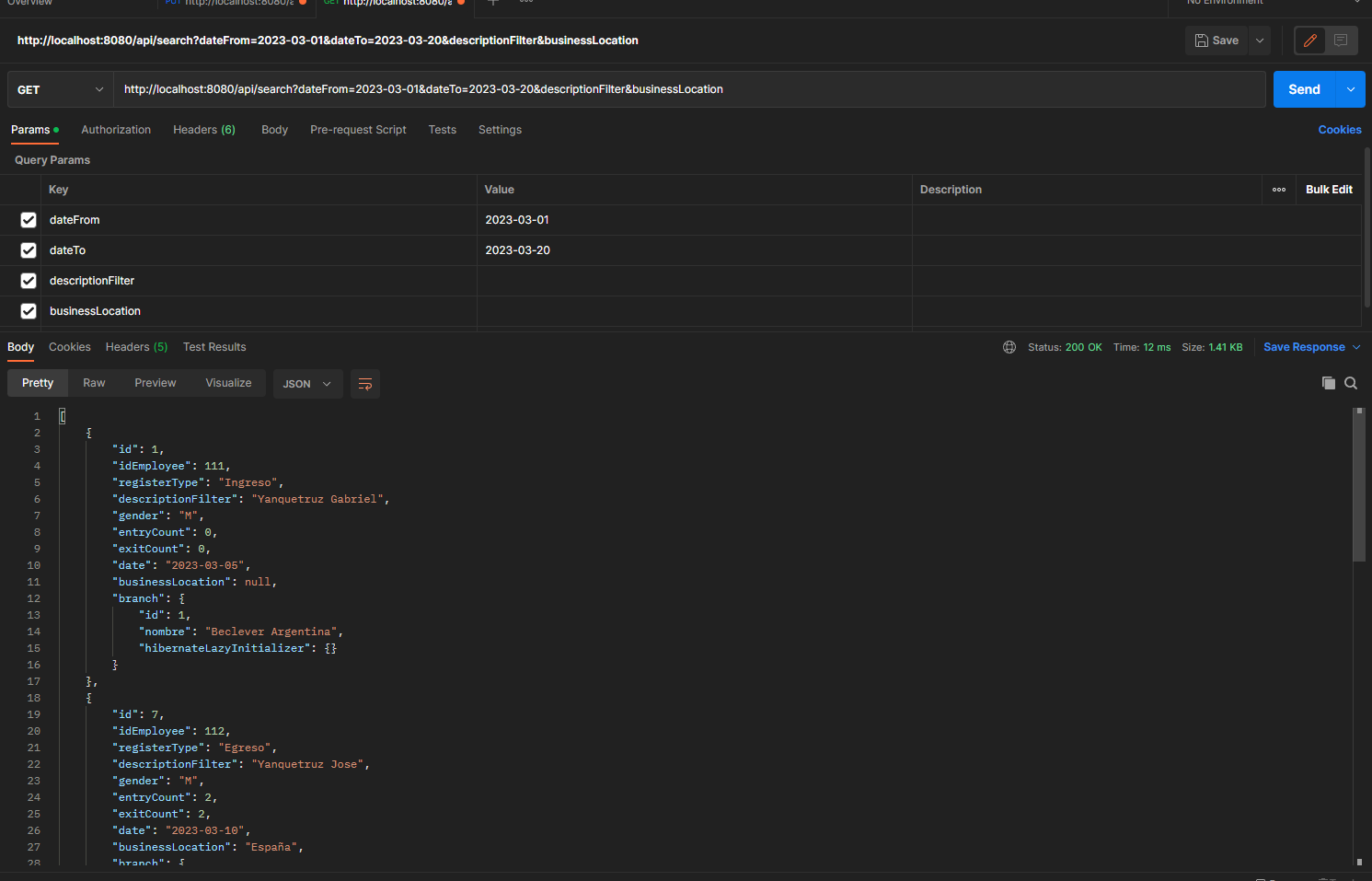
}

} //ID 1 Es beclever argentina - ID 2 es España - 3 es brasil

1. La segunda función es search, para poder buscar y listar la cantidad de ingresos y egresos dada una fecha, con filtrar nombres o apellido y sucursal.

Un ejemplo de la url sería <http://localhost:8080/api/search?dateFrom=2023-03-01&dateTo=2023-03-20&descriptionFilter&businessLocation>

Recibimos un resultado como este.



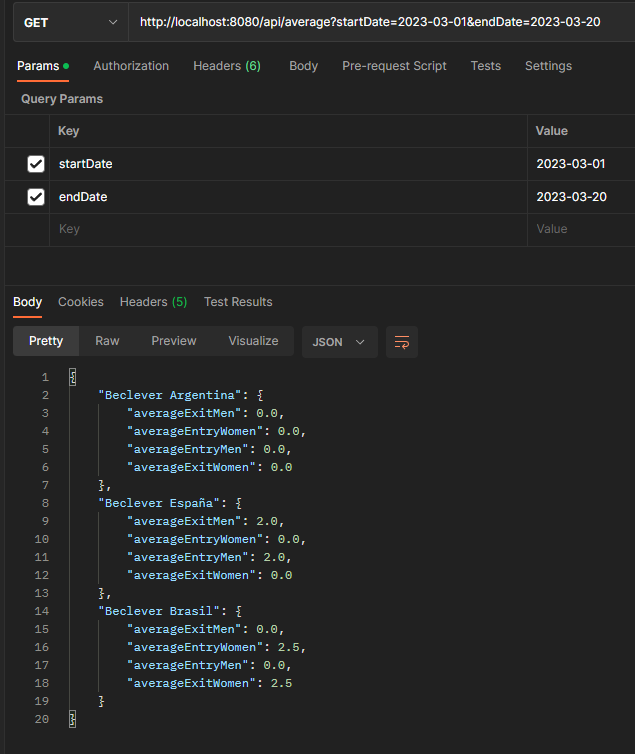
Donde descriptionFilter y businessLocation son opcionales. Las fechas si son obligatorias

6) Por último, la ultima función es generar un servicio que devuelve el promedio de hombres y mujeres que ingresan y egresan por mes por cada sucursal.

En nuestro caso tendremos la función average, siempre debemos recordar respetar el valor de la fecha, es decir “YYYY-MM-DD”

La url de ejemplo seria <http://localhost:8080/api/average?startDate=2023-03-01&endDate=2023-03-20>

Obtendriamos un resultado como este.



Eso sería todo sobre mi desafio, se que puede llegar a tener errores y demás, si es asi mil disculpas.

Un saludo