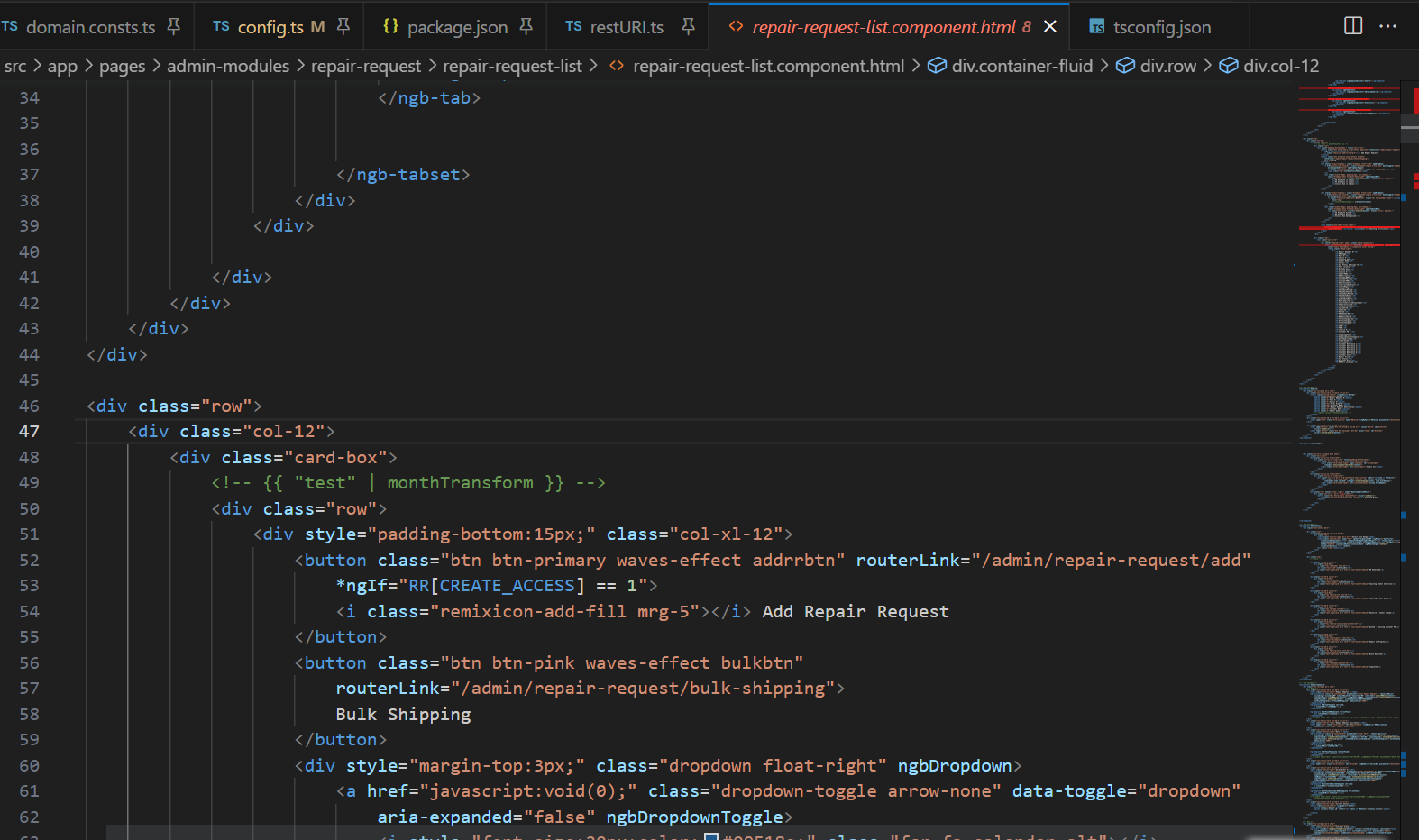
BOC API Document

RR Generated

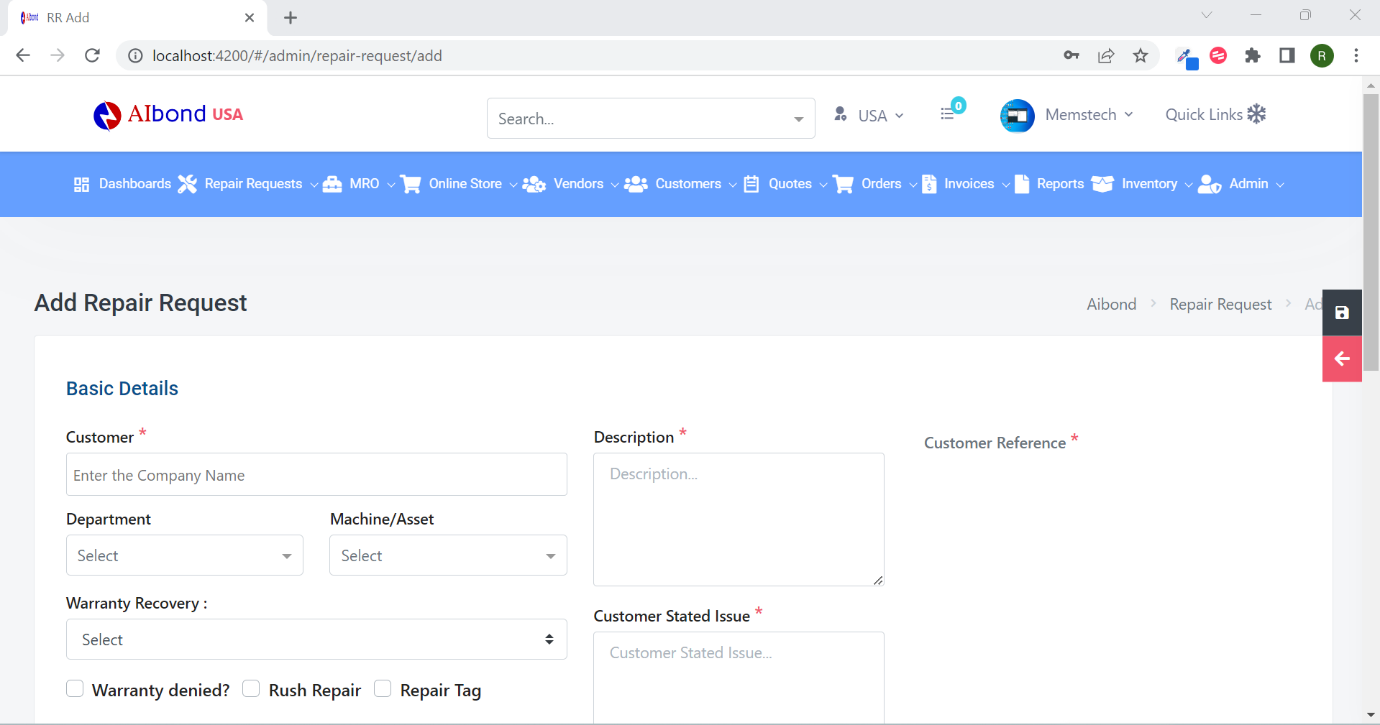
**+Add Repair Request – URL:**

<http://localhost:4200/#/admin/repair-request/add>

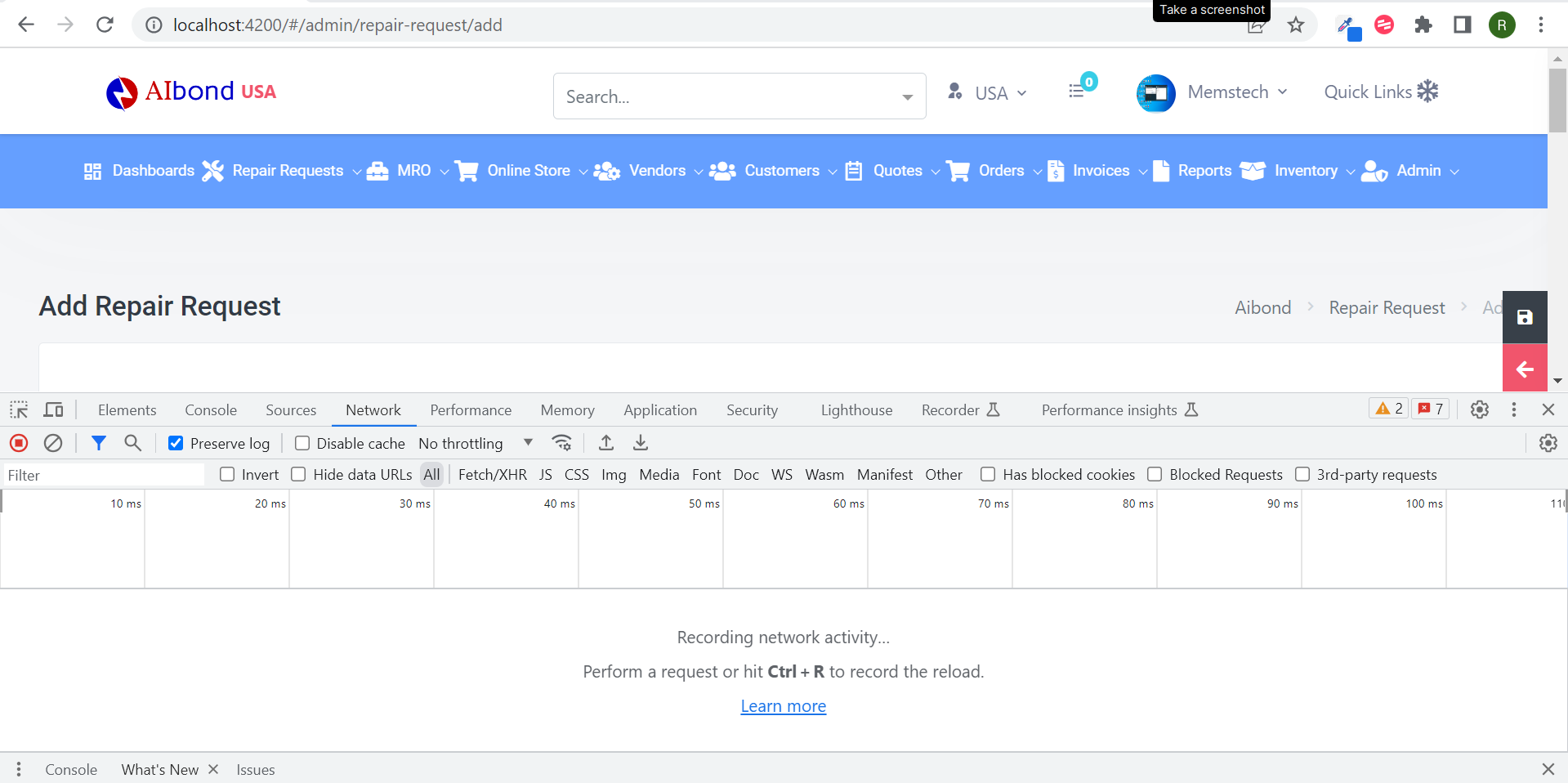


Line 52 shows the router link for “+Add Repair Request” in “repair-request-list.component.html” file.

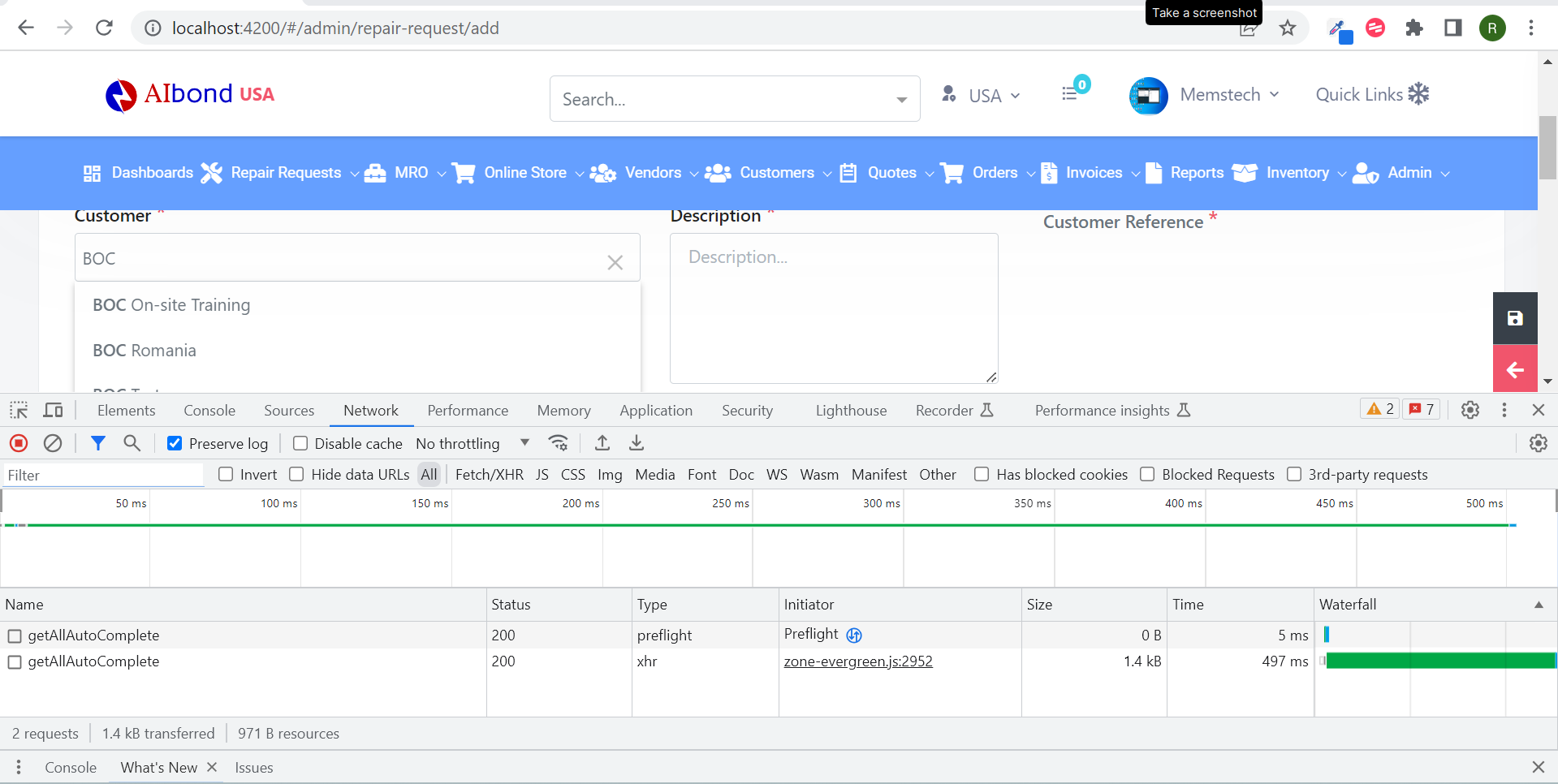
Click “Enter the Company Name” field:



Inspect “Enter the Company Name” field:

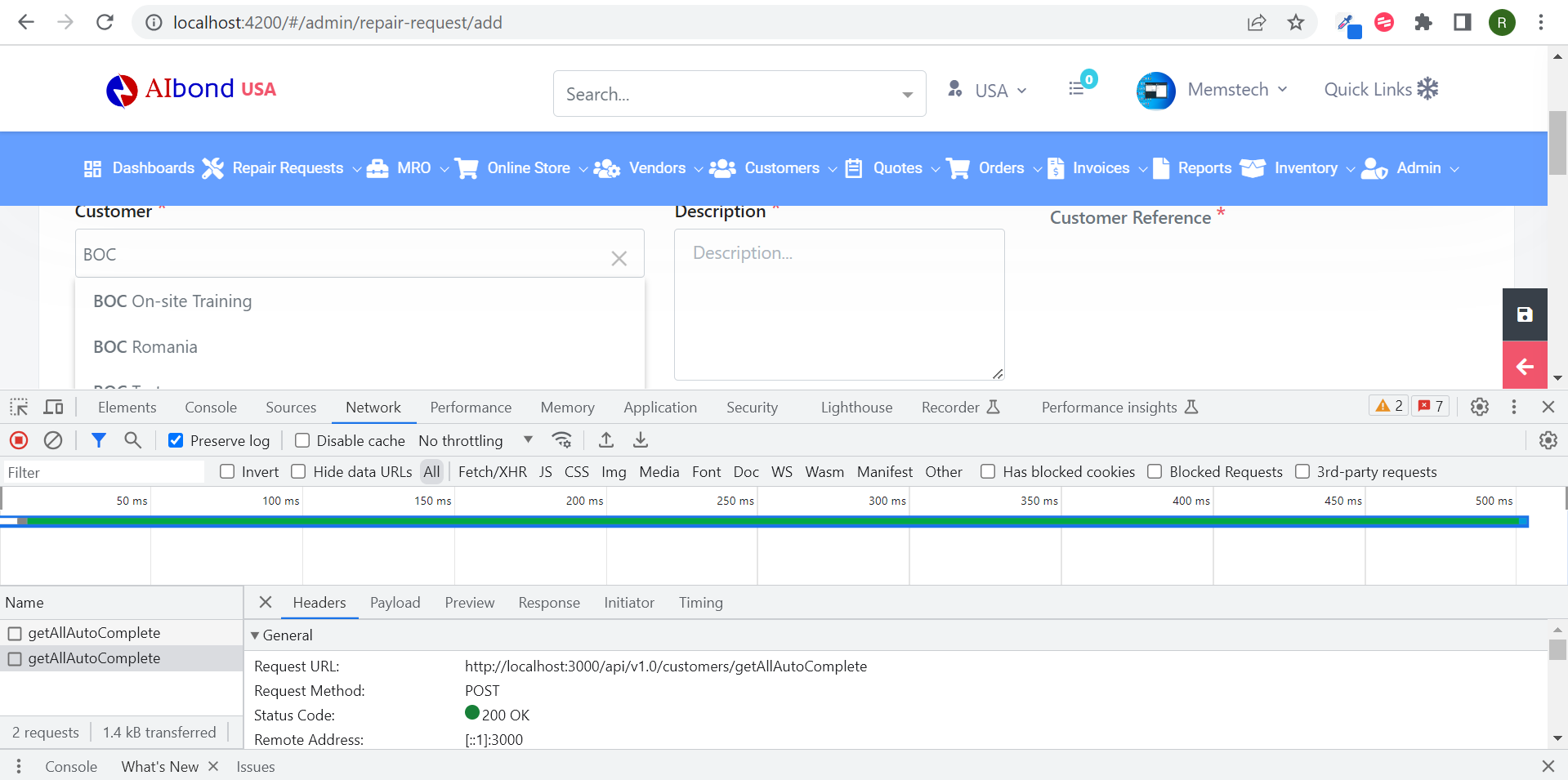


Type “BOC”:



The API for “Customer” is:

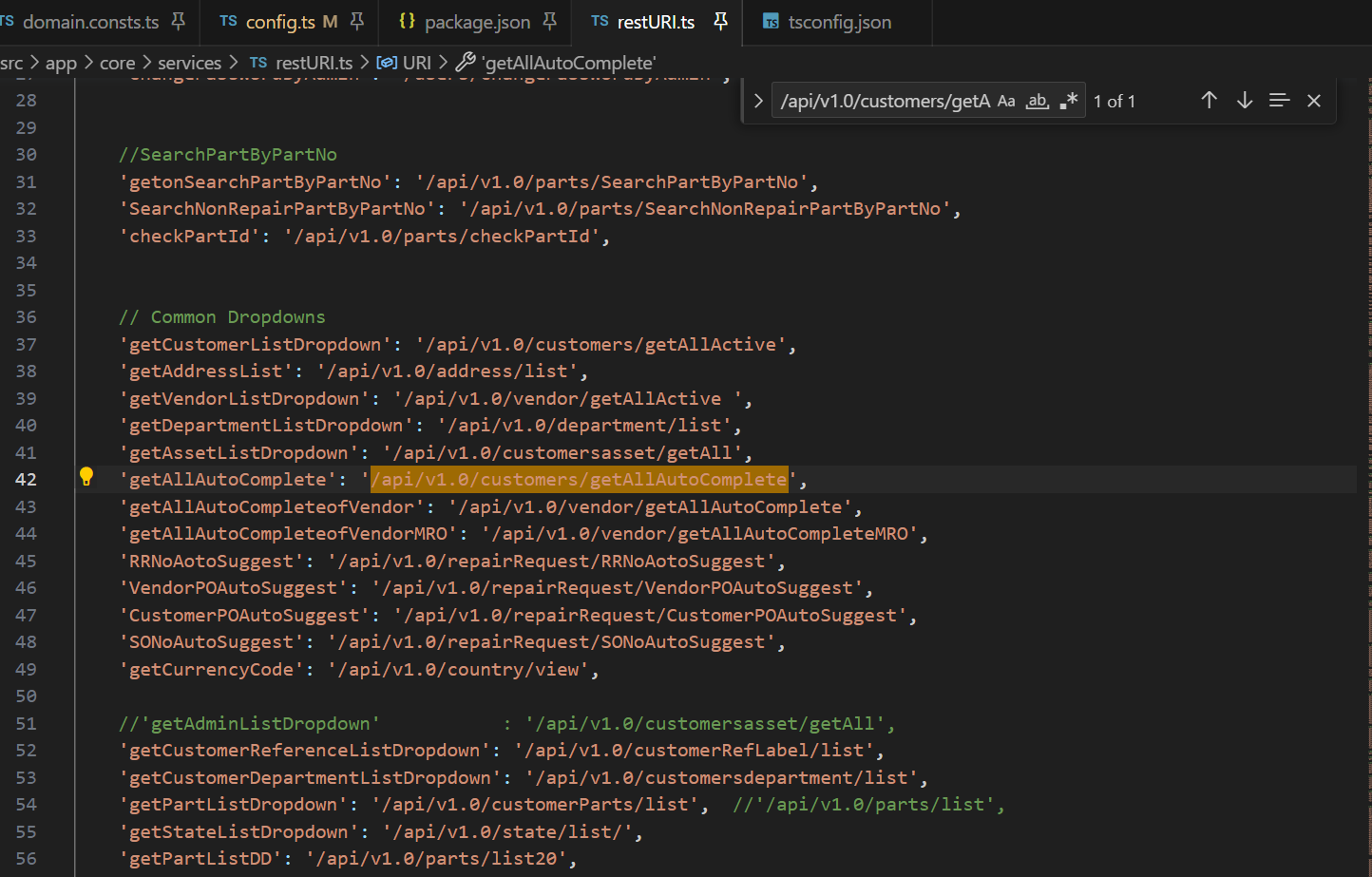
* “/api/v1.0/customers/getAllAutoComplete”



1. **“/api/v1.0/customers/getAllAutoComplete”**

Open “Front End” codebase”:

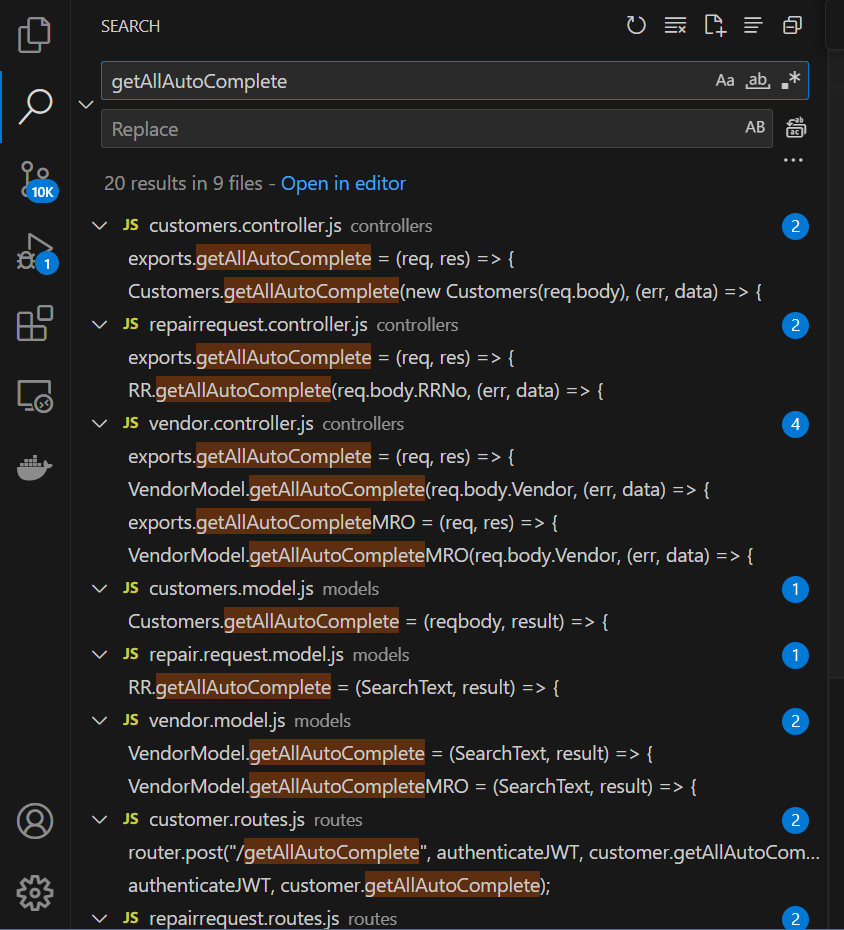
Search the “Customer” API in “restURI.ts” file:



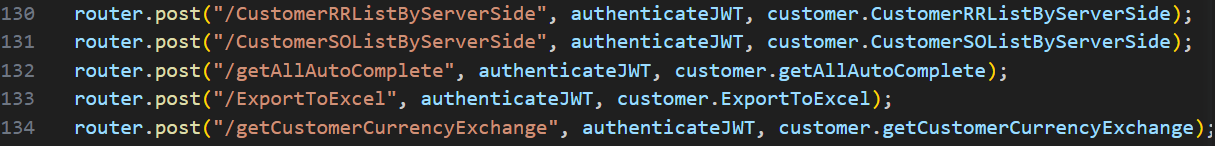
Copy: “getAllAutoComplete” to search in frontend.

Open “Back End” codebase:

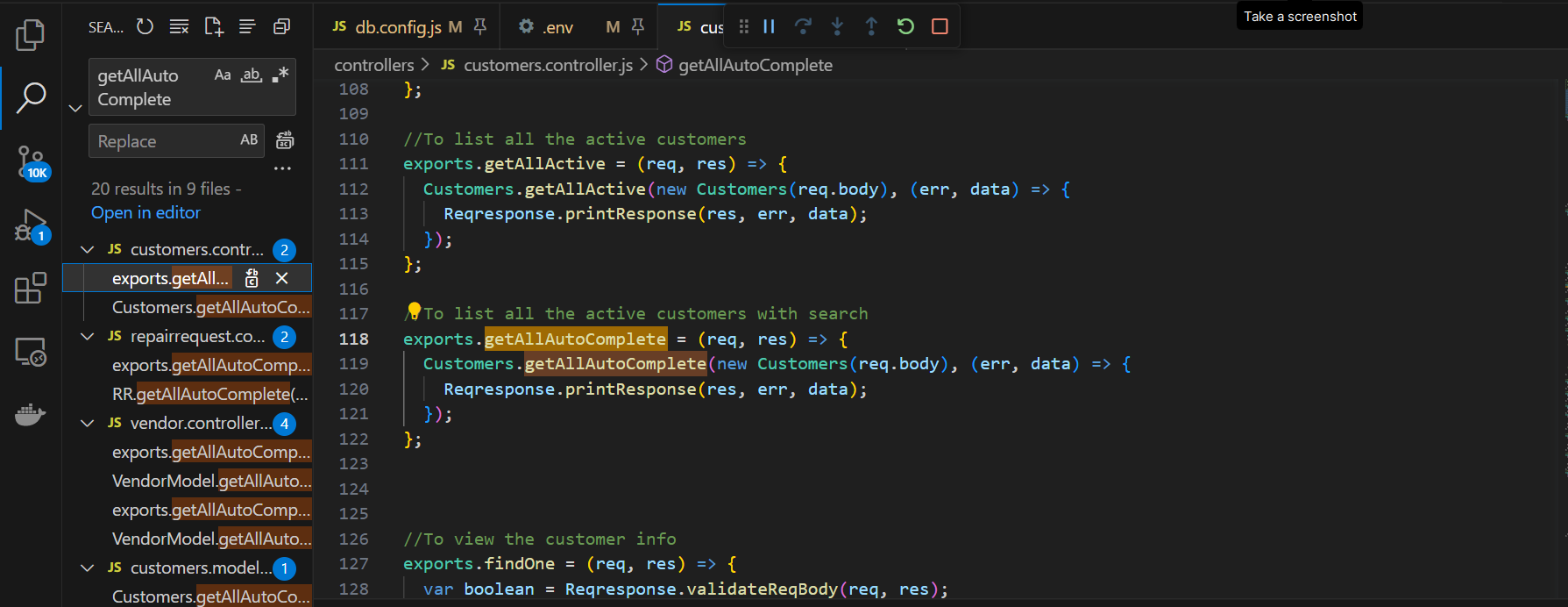
Search “getAllAutoComplete” in the “Search” bar,



Open “customer.routes.js” file,

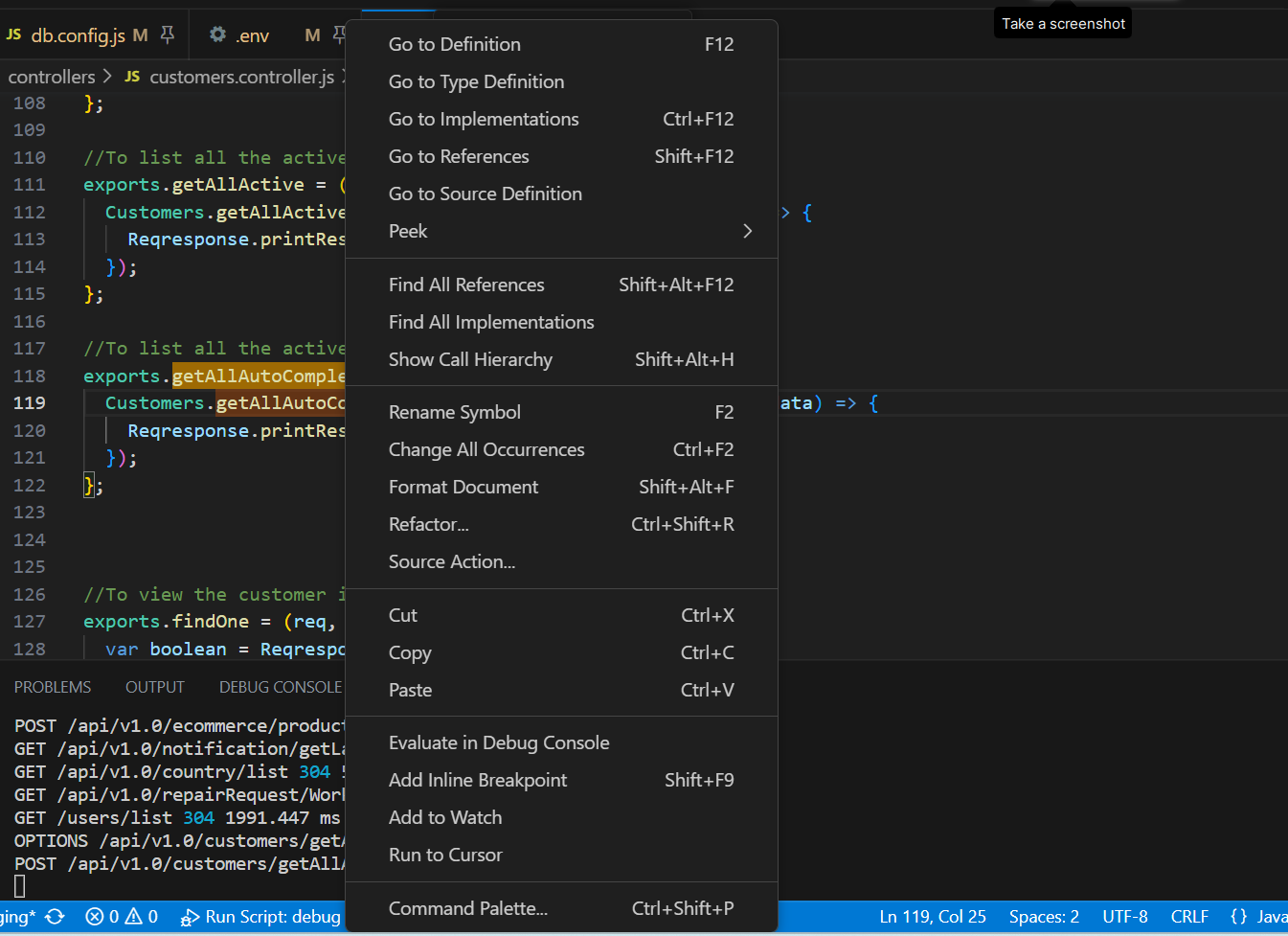


“Right” click on “customer.getAllAutoComplete”, in line 132, click “Go to Definition”, it shows the code in “customers.controller.js”,

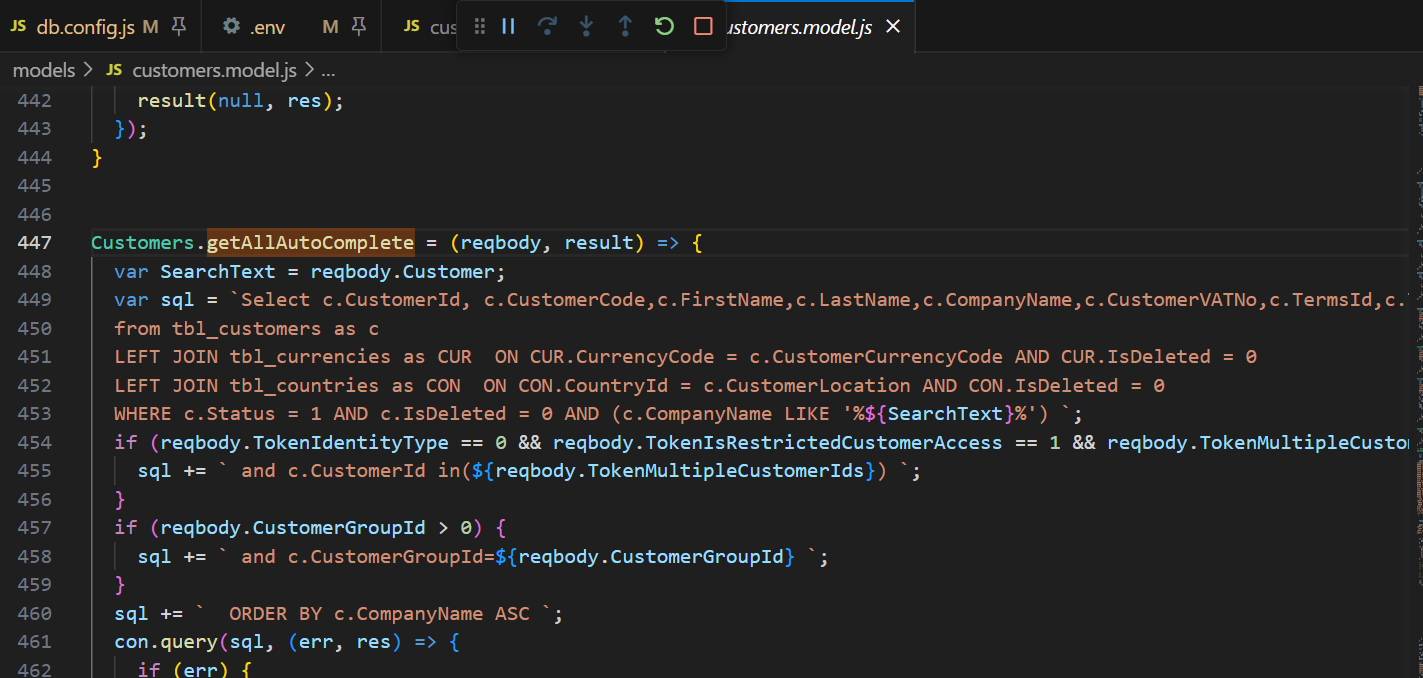


Line 118 shows its code.

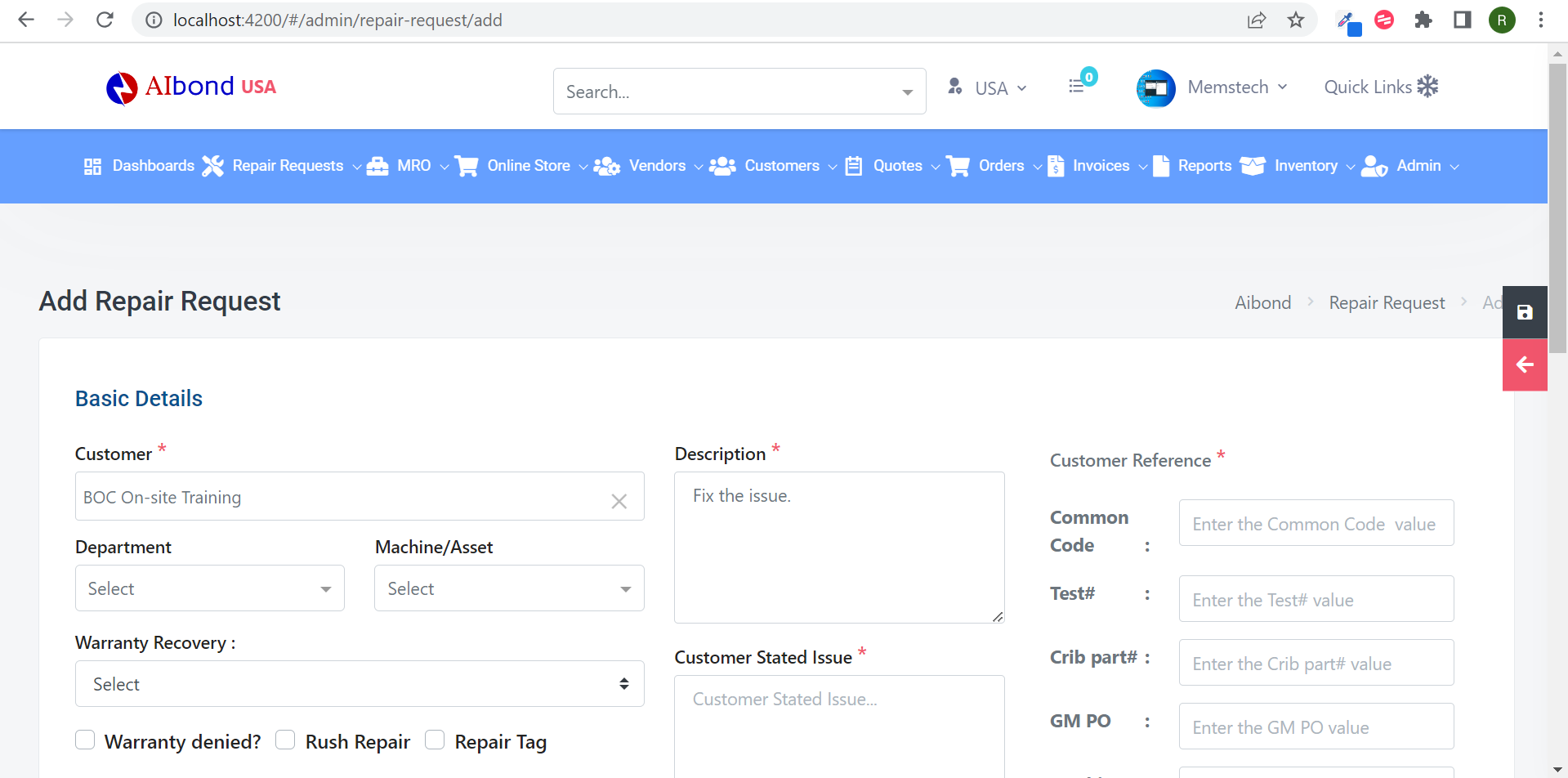
“Right” click “getAllAutoComplete” in line 119,



Click “Go to Definition”,

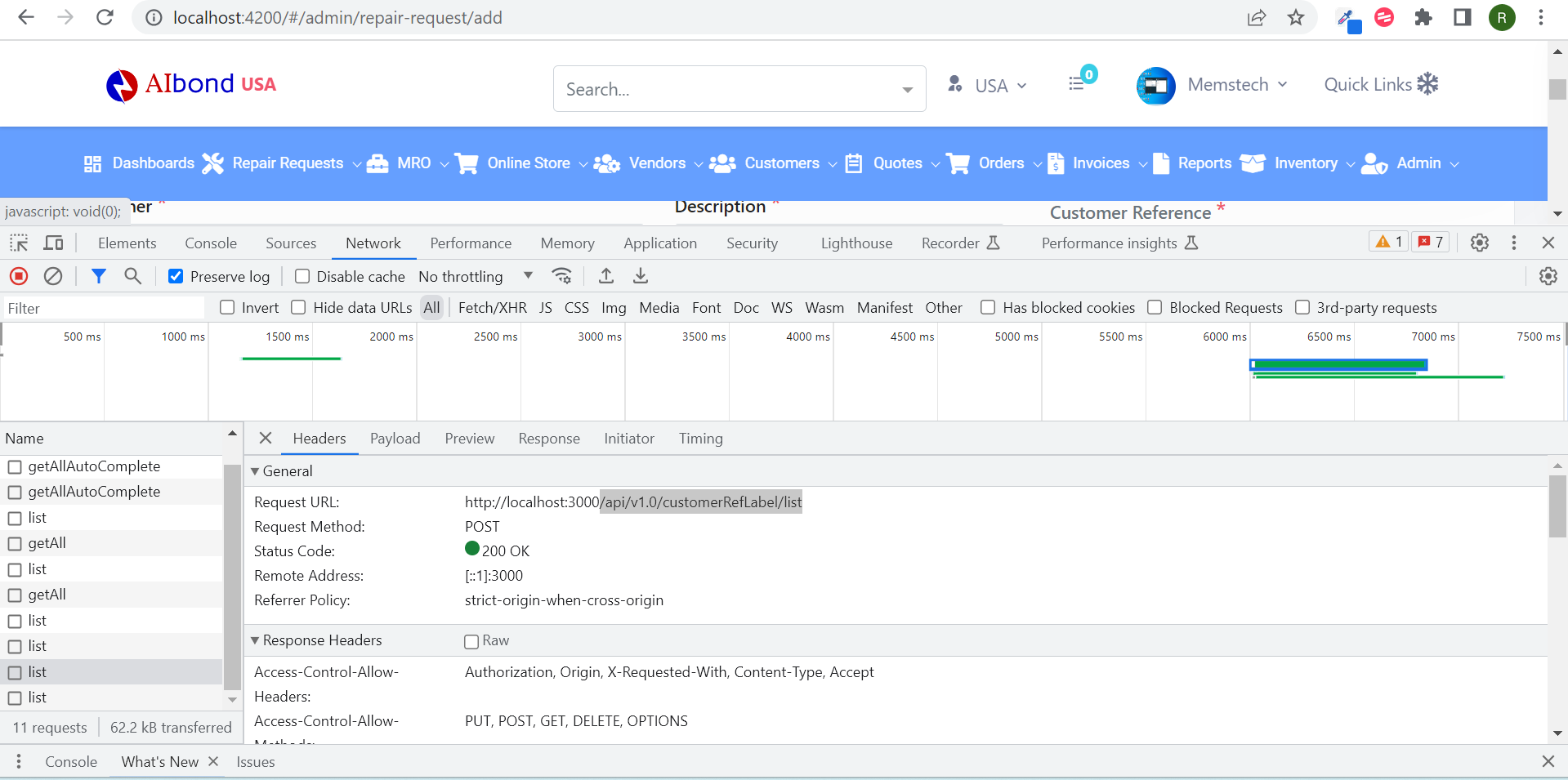


It goes to “customers.model.js” and shows its “SQL” codebase in line 447.



Now, “Customer Reference” dropdown displays.

Inspect the “Customer” field”, click “BOC On-site Training” from the “Customer” option,



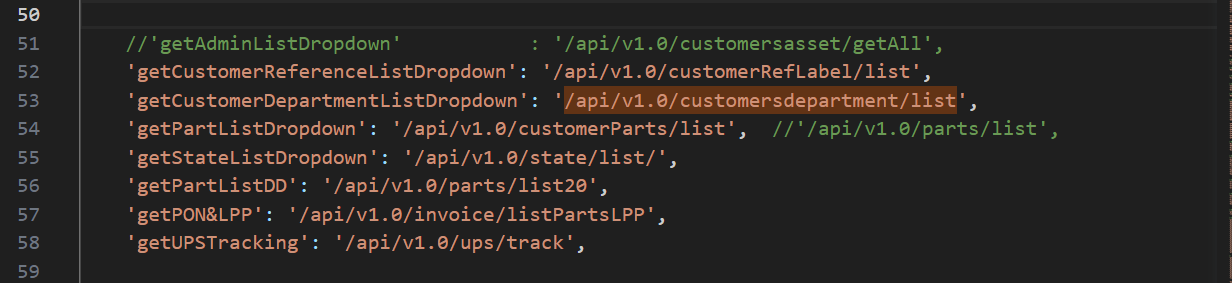
API’s:

* /api/v1.0/customersdepartment/list
* /api/v1.0/customersasset/getAll
* /api/v1.0/customerRefLabel/list
* /api/v1.0/customerParts/list

1. **“/api/v1.0/customersdepartment/list”**

Open “Front End” codebase”:

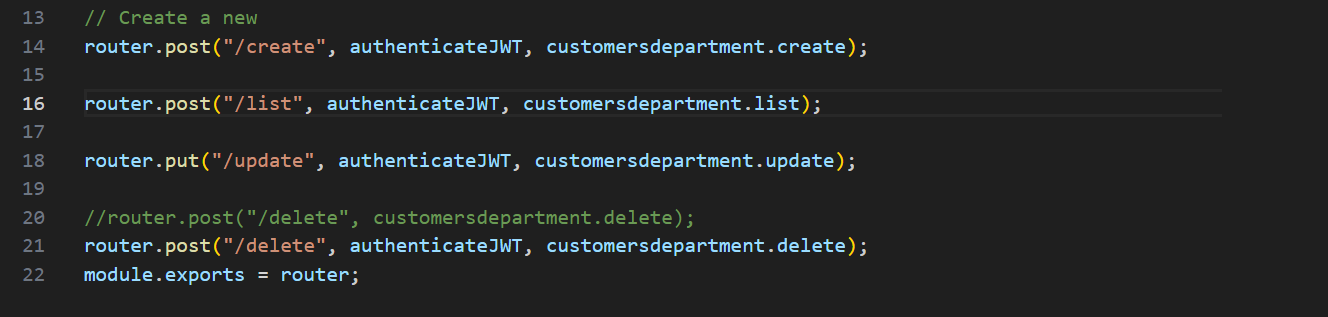
Search the API in “restURI.ts” file:



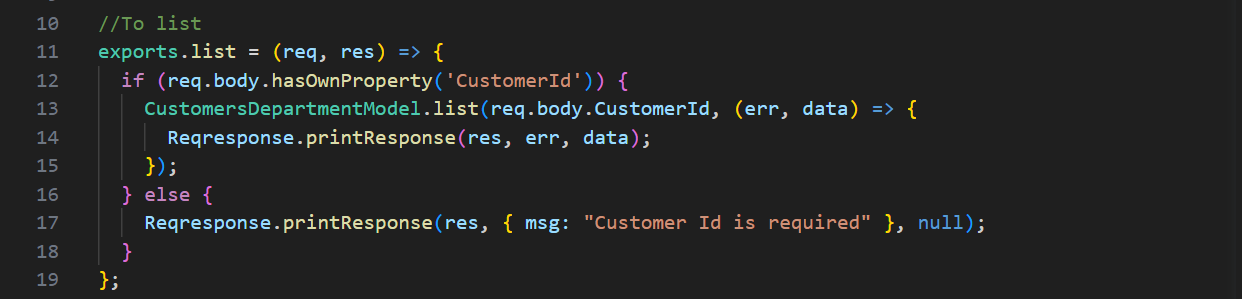
Copy: “getCustomerDepartmentListDropdown” to search in frontend.

Open “Back End” codebase:

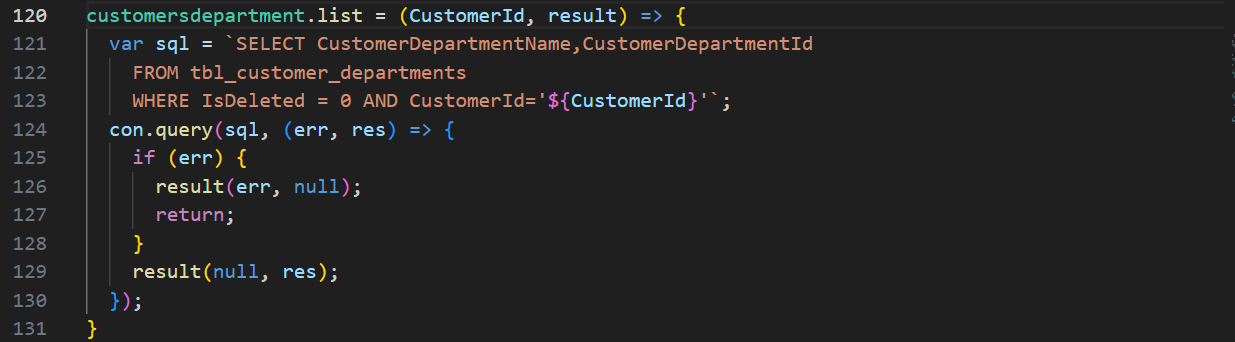
Open “customersdepartment.routes.js”,



“Right” click on “customersdepartment.list” in line 16, click “Go to Definition”, it shows the code in “customersdepartment.controller.js”,



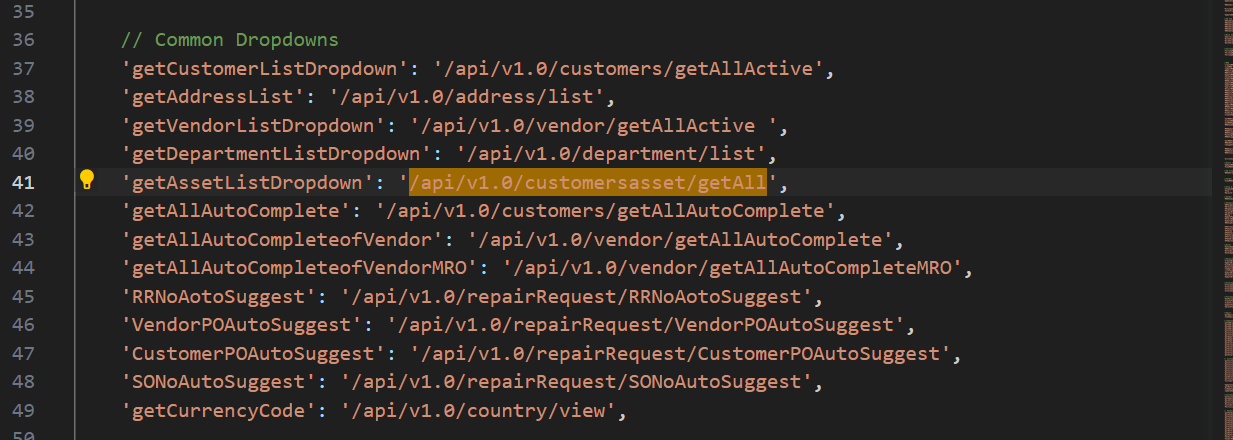
“Right” click on “list” in line 13, click “Go to Definition”, it shows its “SQL” codebase in “customersdepartment.model.js” in line 120,



1. **“/api/v1.0/customersasset/getAll”**

Open “Front End” codebase”:

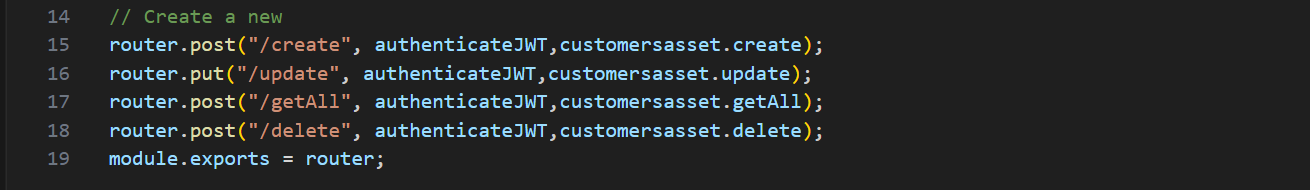
Search the API in “restURI.ts” file:



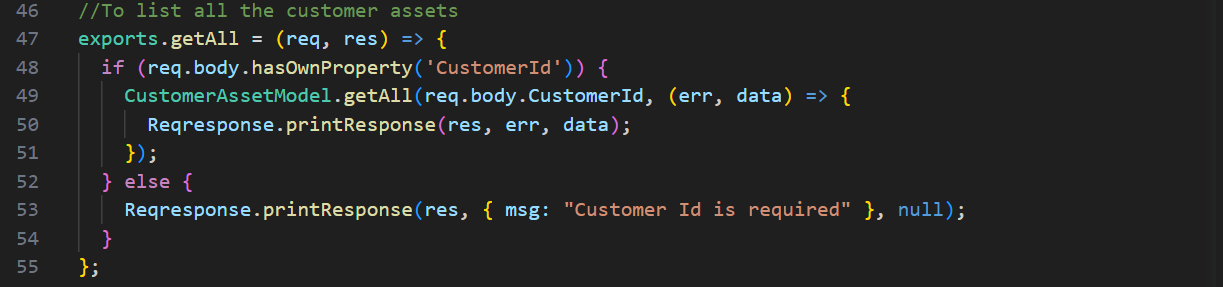
Copy: “getAssetListDropdown” to search in frontend.

Open “Back End” codebase:

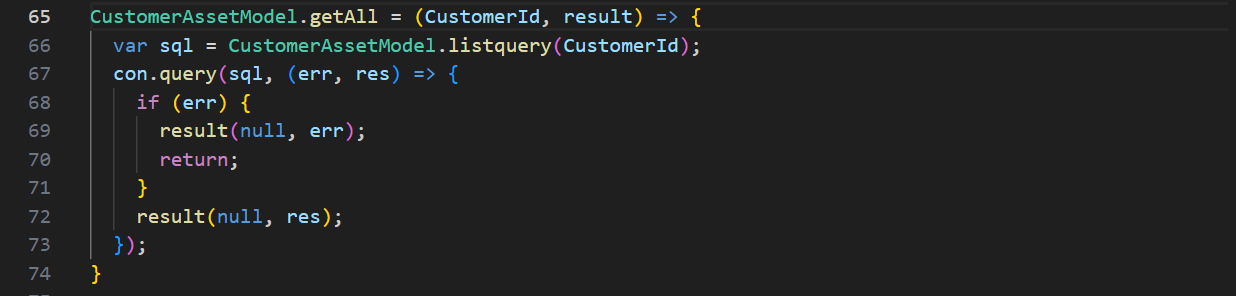
Open “customersassest.routes.js”,



“Right” click on “customersassest.getAll” in line 17, click “Go to Definition”, it shows the code in “customersassest.controller.js”,



“Right” click on “getAll” in line 49, click “Go to Definition”, it shows its “SQL” codebase in “customersassest.model.js”,



1. **“/api/v1.0/customerRefLabel/list”**

Open “Front End” codebase”:

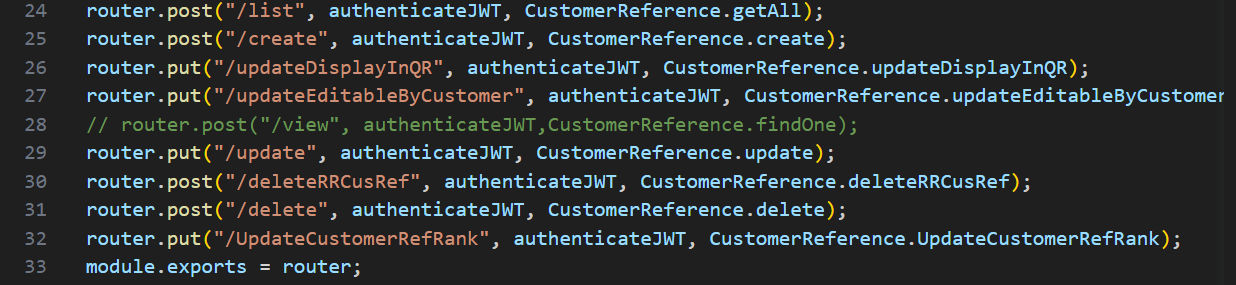
Search the API in “restURI.ts” file:



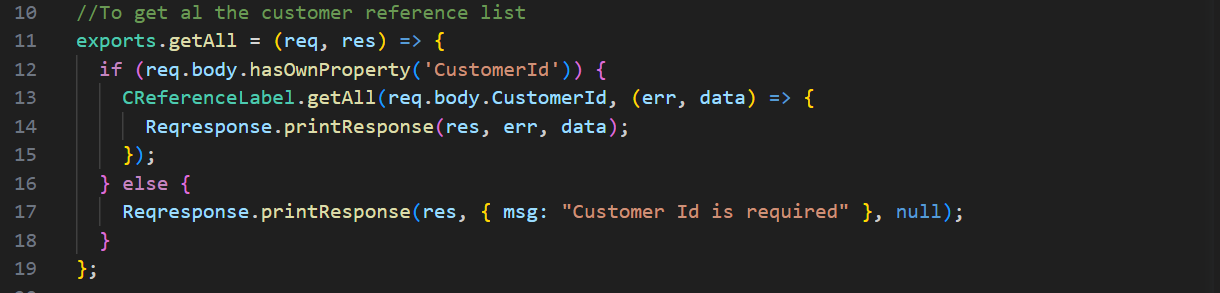
Copy: “getCustomerReferenceListDropdown” to search in frontend.

Open “Back End” codebase:

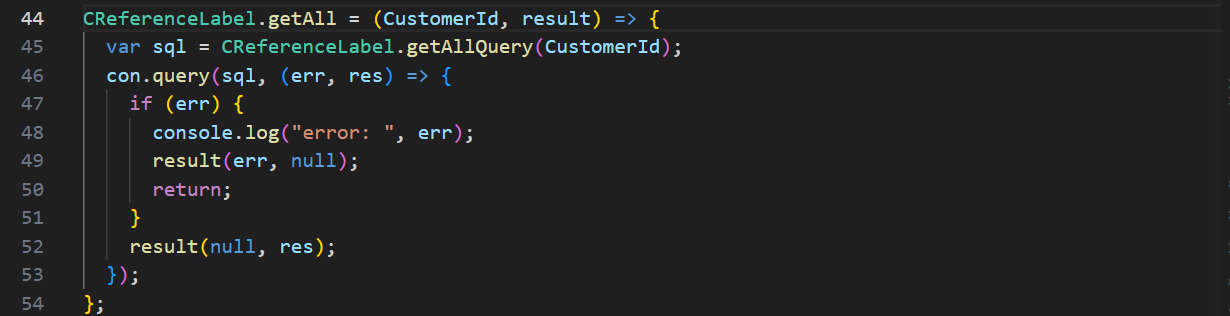
Open “customer.reference.labels.routes.js”,



“Right” click on “customerReference.getAll” in line 24, click “Go to Definition”, it shows the code in “customer.reference.label.controller.js”,



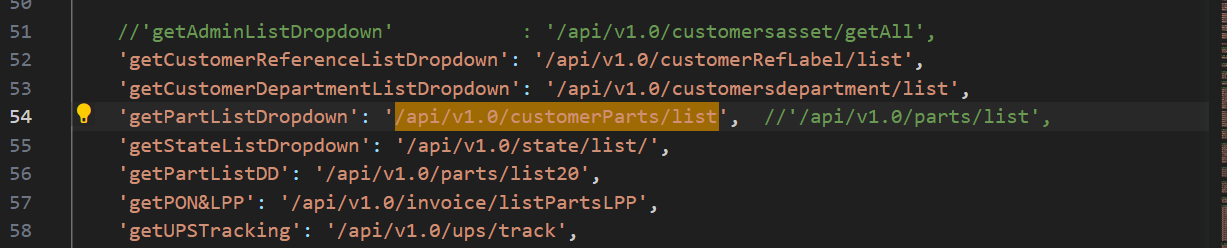
“Right” click on “getAll” in line 13, click “Go to Definition”, it shows its “SQL” codebase in “customer.reference.labels.model.js”,



1. **“/api/v1.0/customerParts/list”**

Open “Front End” codebase”:

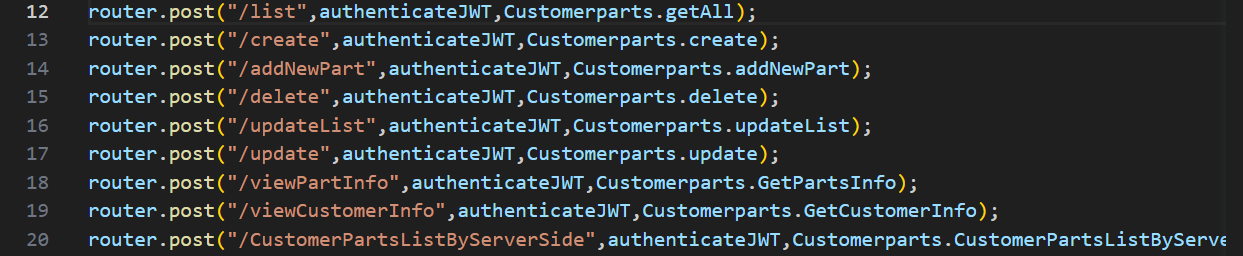
Search the API in “restURI.ts” file:



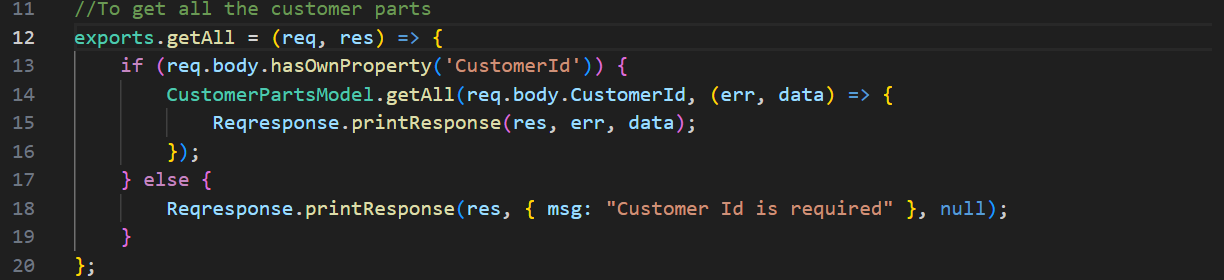
Copy: “getPartListDropdown” to search in frontend.

Open “Back End” codebase:

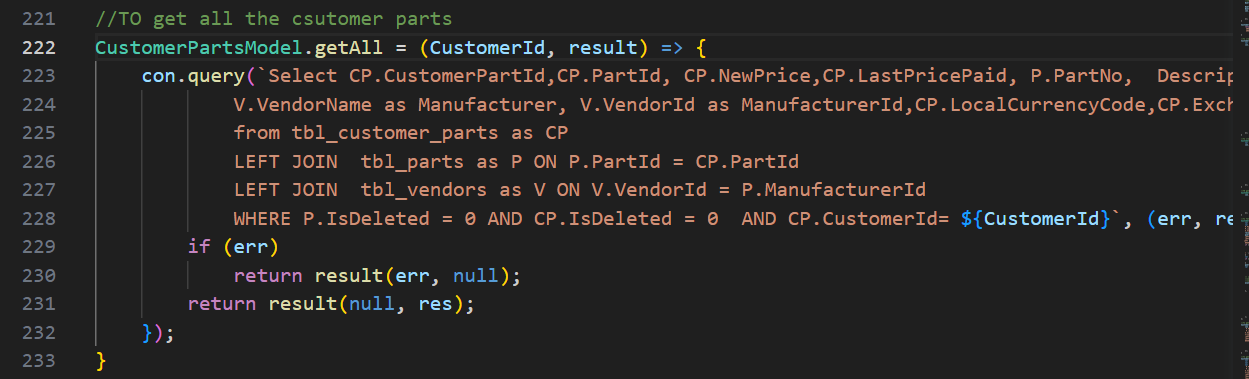
Open “customer.parts.routes.js”,



“Right” click on “customerparts.getAll” in line 12, click “Go to Definition”, it shows the code in “customer.parts.controller.js”,



“Right” click on “getAll” in line 14, click “Go to Definition”, it shows its “SQL” codebase in “customer.parts.model.js”,



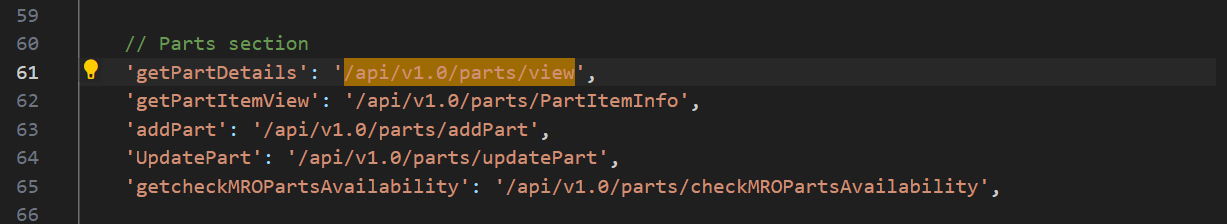
By clicking the “Select Part#” field:

* /api/v1.0/parts/view
* /api/v1.0/invoice/listPartsLPP

1. **“/api/v1.0/parts/view”**

Open “Front End” codebase”:

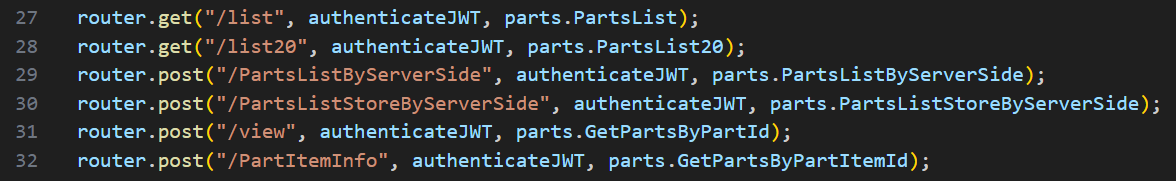
Search the “Customer Reference” API in “restURI.ts” file:



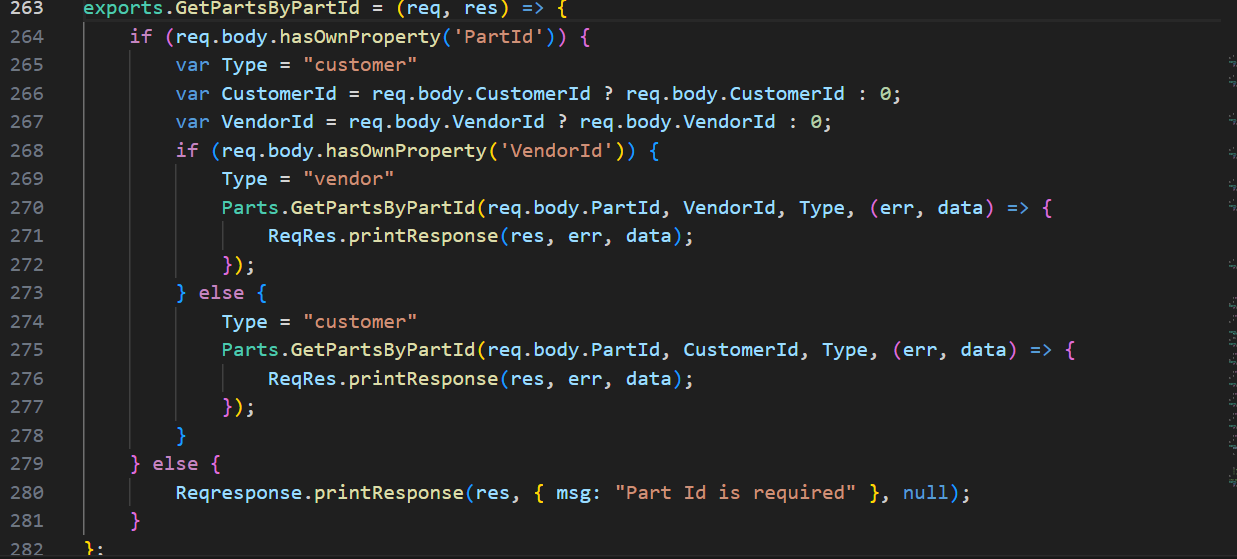
Copy: “getPartDetails” to search in frontend.

Open “Back End” codebase:

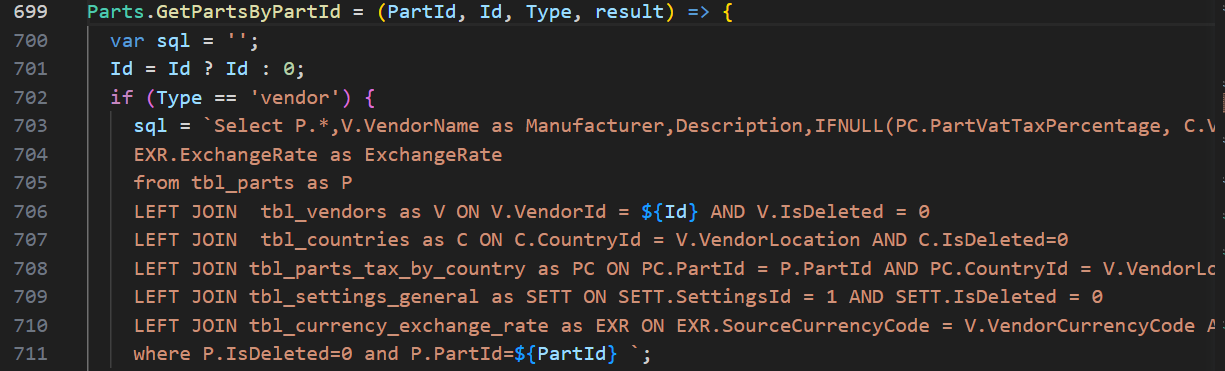
Open “parts.routes.js”,

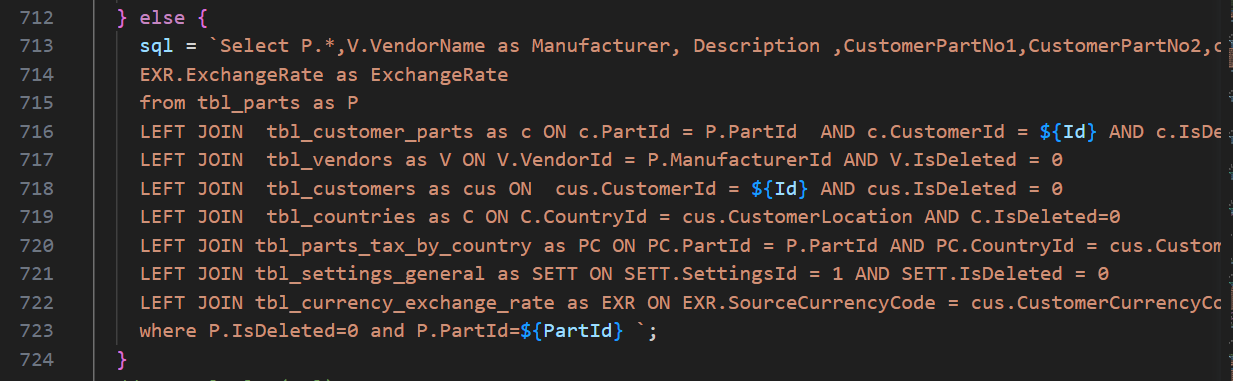


“Right” click on “parts.GetPartsByPartId” in line 31, click “Go to Definition”, it shows the code in “parts.controller.js”,



“Right” click on “GetPartsByPartId” in line 275, click “Go to Definition”, it shows its “SQL” codebase in “parts.model.js”,



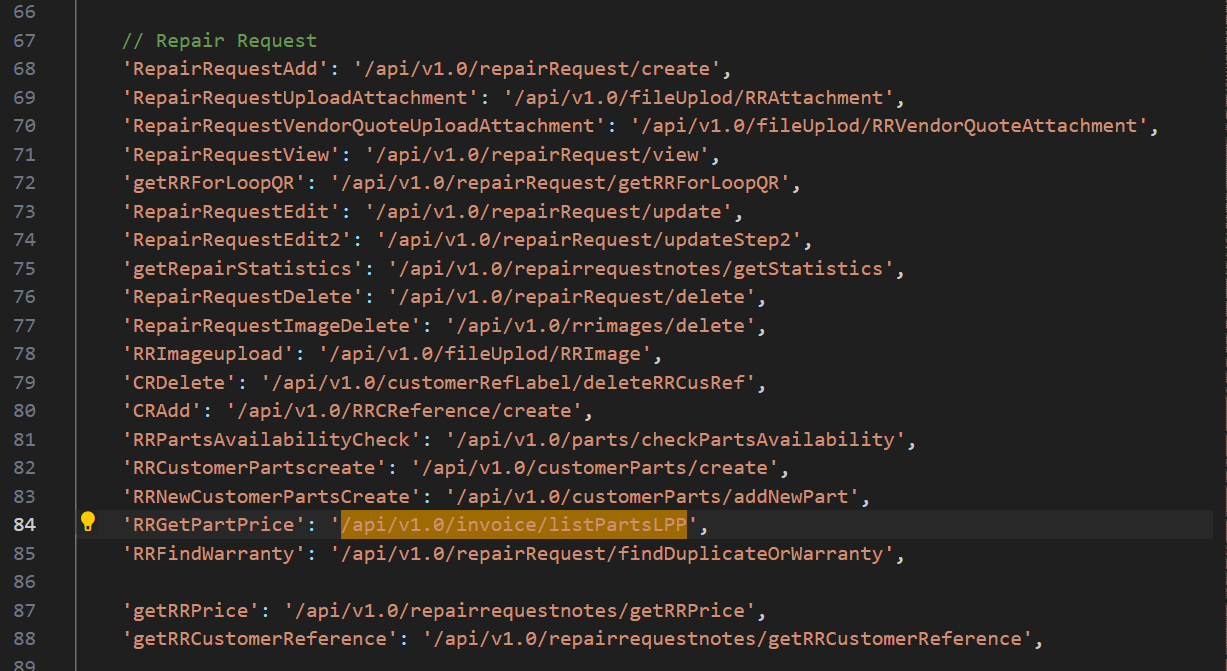




1. **“/api/v1.0/invoice/listPartsLPP”**

Open “Front End” codebase”:

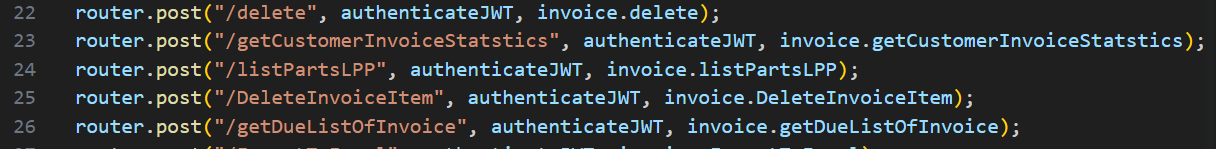
Search the “Customer Reference” API in “restURI.ts” file:



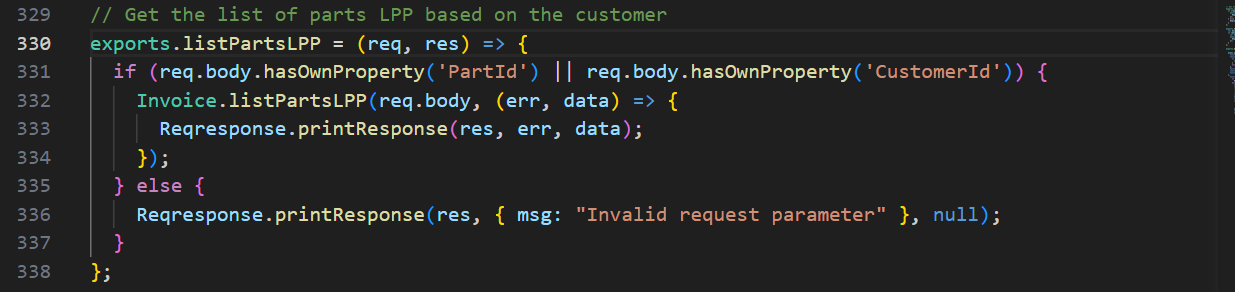
Copy: “RRGetPartPrice” to search in frontend.

Open “Back End” codebase:

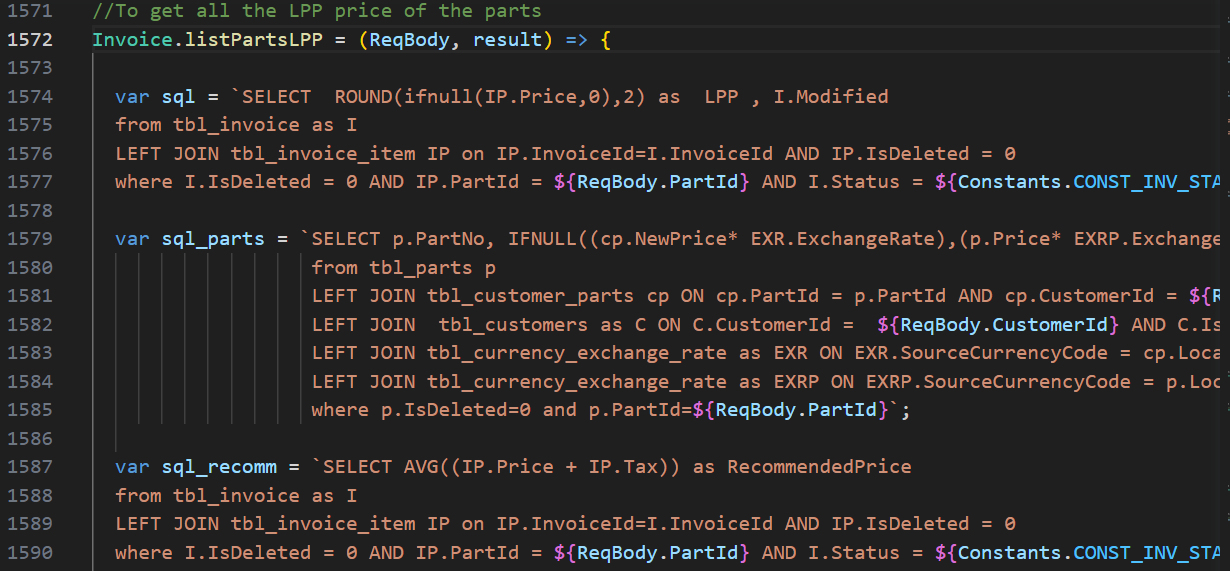
Open “invoice.routes.js”,

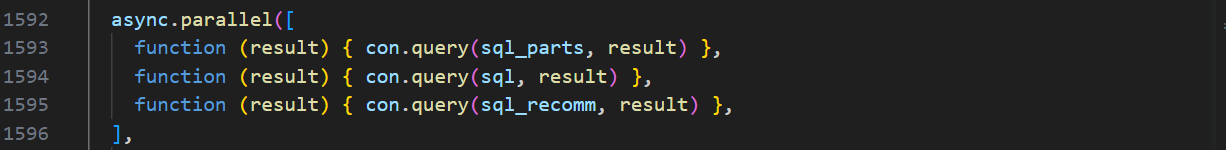


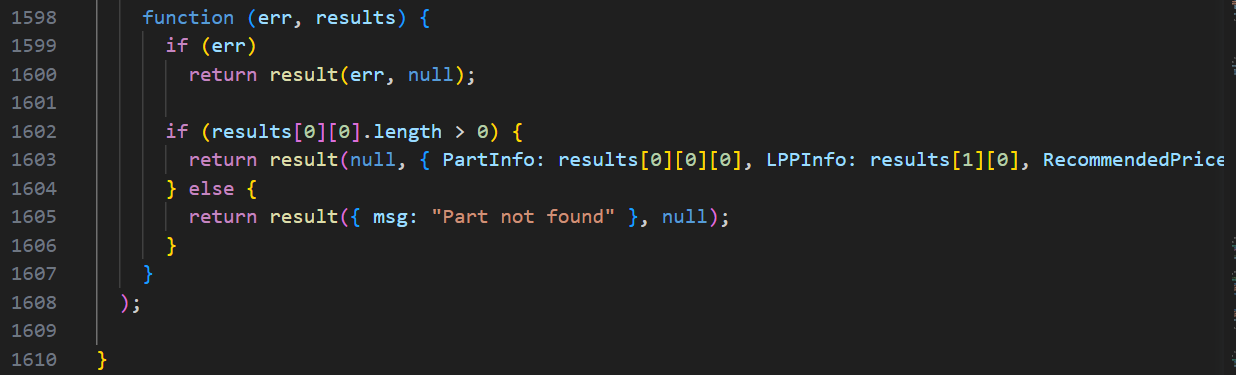
“Right” click on “invoice.listPartsLPP” in line 24, click “Go to Definition”, it shows the code in “invoice.controller.js”,



“Right” click on “listPartsLPP” in line 332, click “Go to Definition”, it shows its “SQL” codebase in “invoice.model.js”,







By clicking the “Serial#” button:

* / api/v1.0/repairRequest/findDuplicateOrWarranty

1. **“/api/v1.0/repairRequest/findDuplicateOrWarranty”**

Open “Front End” codebase”:

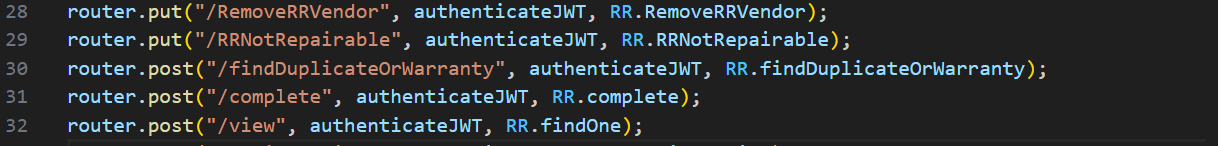
Search the “Customer Reference” API in “restURI.ts” file:



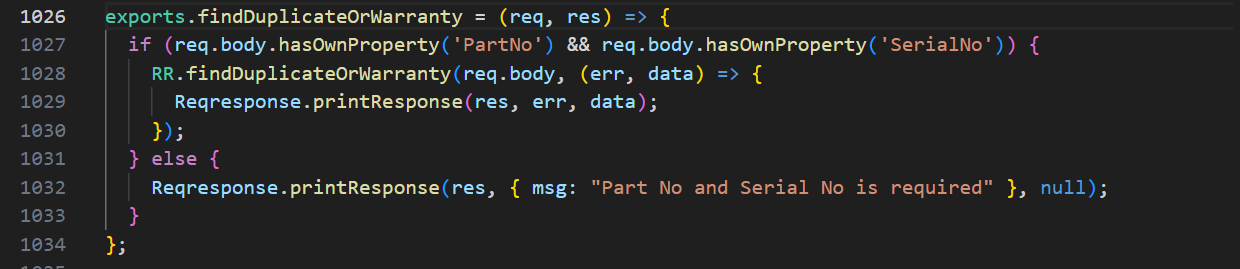
Copy: “RRFindWarranty” to search in frontend.

Open “Back End” codebase:

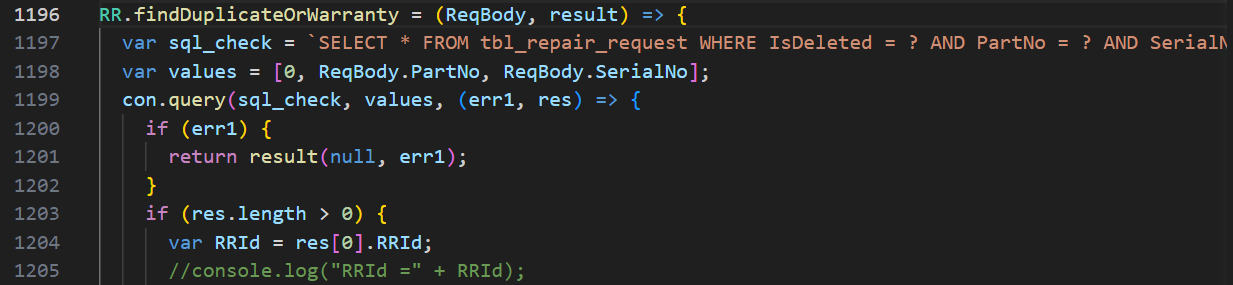
Open “repairrequest.routes.js”,

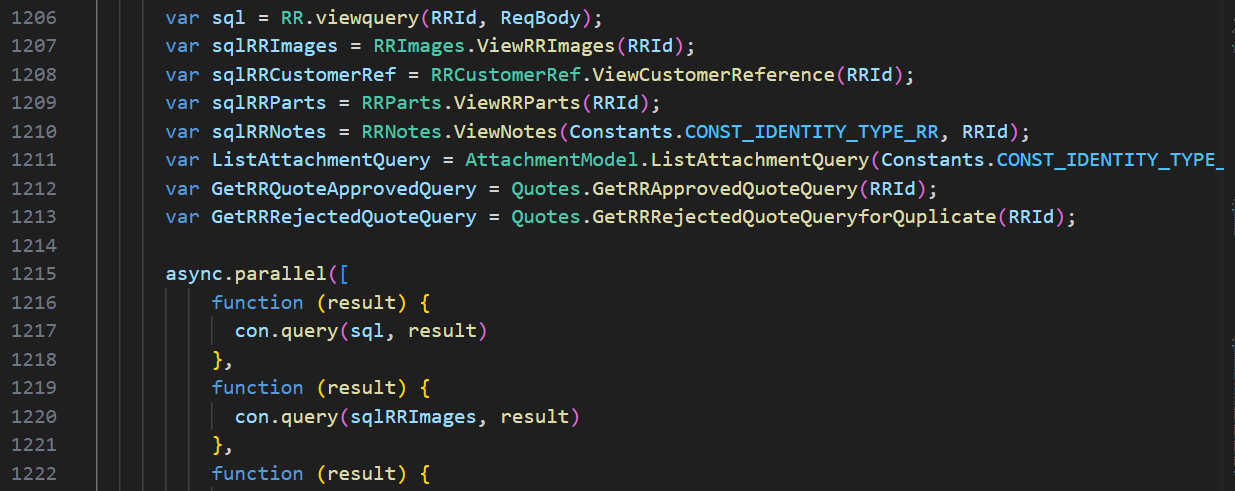


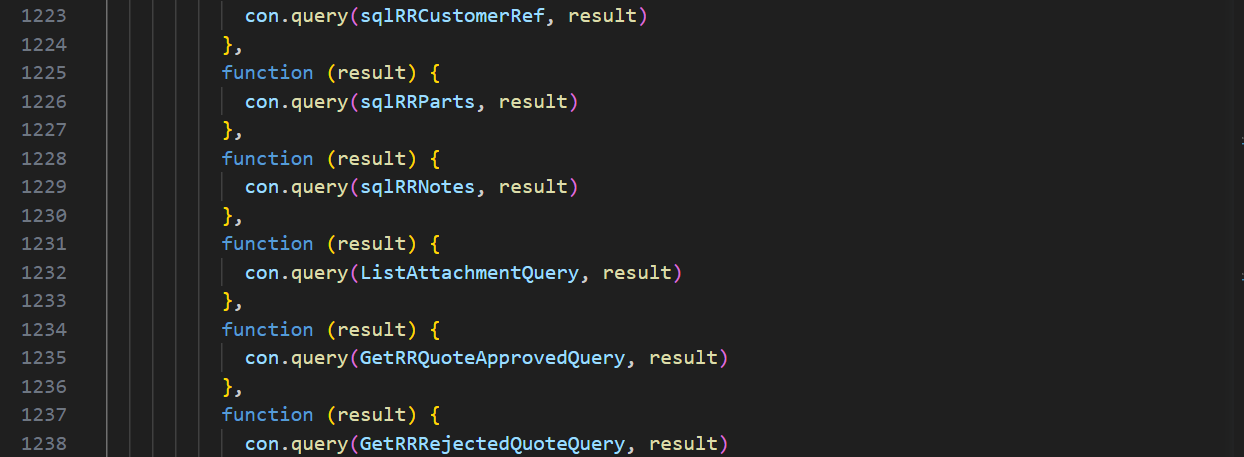
“Right” click on “RR.findDuplicateOrWarranty” in line 30, click “Go to Definition”, it shows the code in “repairrequest.controller.js”,

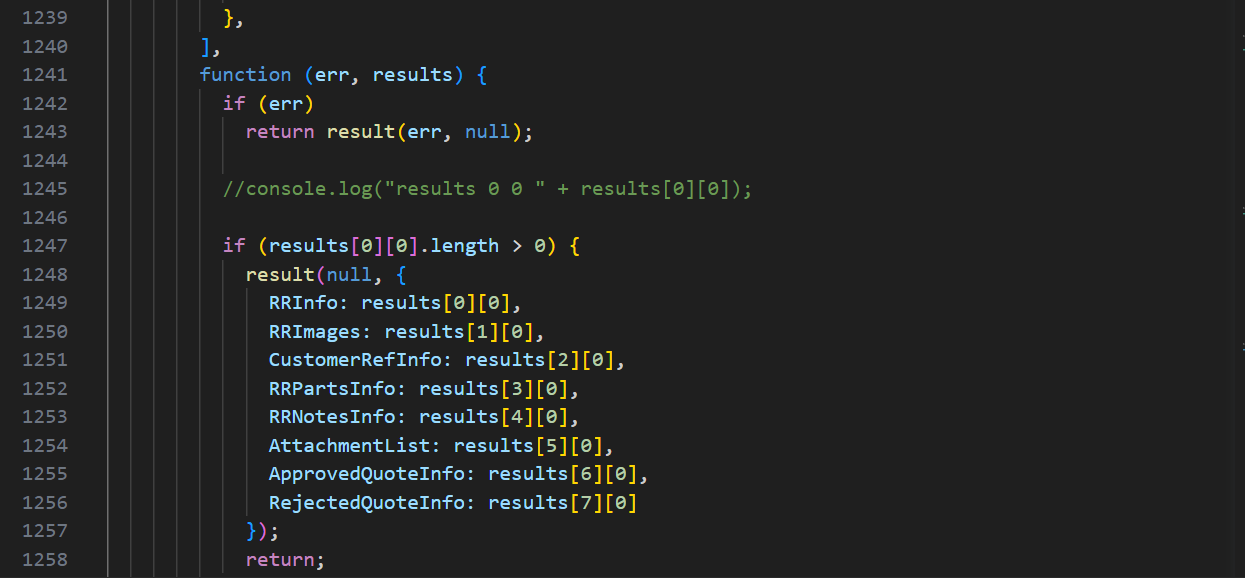


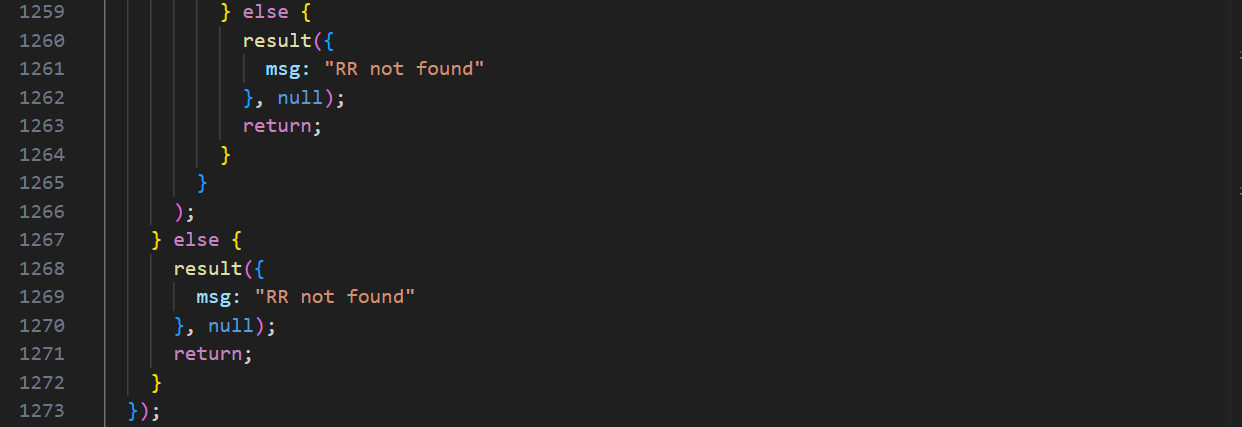
“Right” click on “findDuplicateOrWarranty” in line 1028, click “Go to Definition”, it shows its “SQL” codebase in “repairrequest.model.js”,











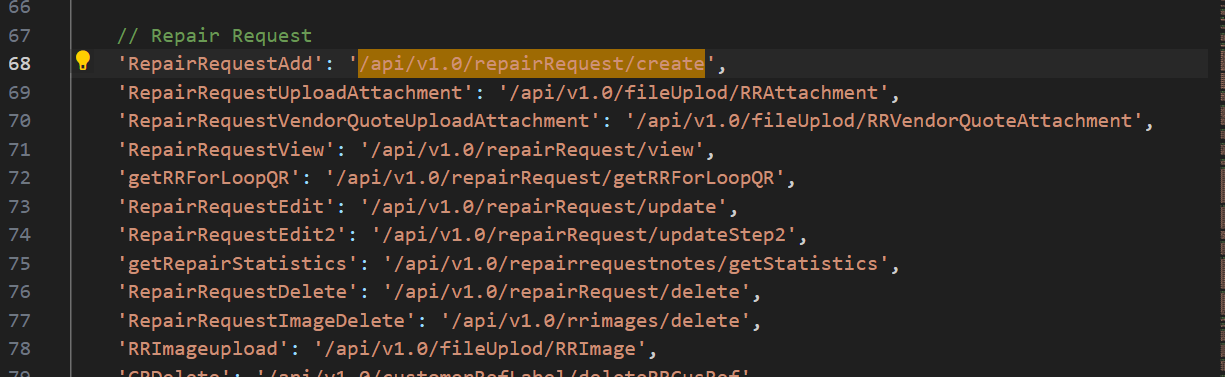
By clicking the save button:

* /api/v1.0/repairRequest/create
* /api/v1.0/notification/getLatest
* /api/v1.0/repairRequest/WorkchainMyTaskCount
* /api/v1.0/ecommerce/product/cartCount
* /api/v1.0/country/list
* /api/v1.0/repairRequest/view
* /users/getAllActiveAdmin
* /api/v1.0/RRSubStatus/dropdown
* /api/v1.0/currencyExchangeRate/exchange
* /api/v1.0/customerParts/list

1. **“/api/v1.0/repairRequest/create”**

Open “Front End” codebase”:

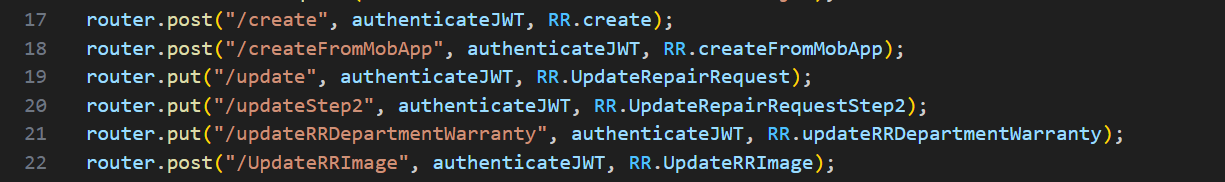
Search the “Customer Reference” API in “restURI.ts” file:



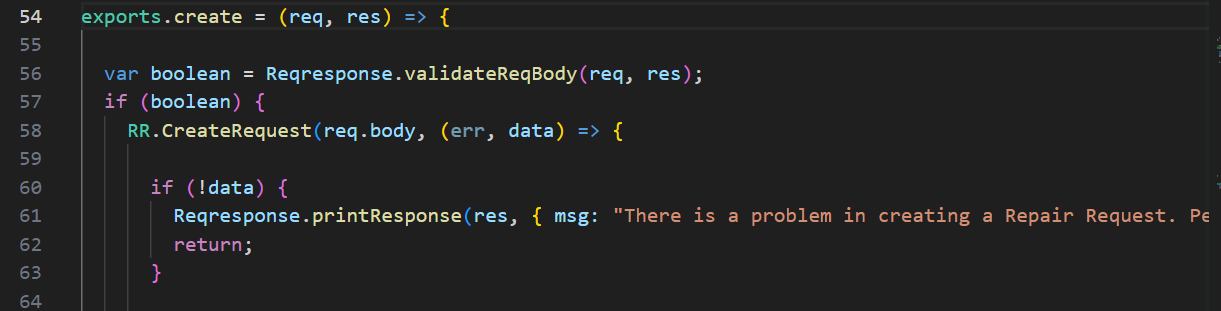
Copy: “RepairRequestAdd” to search in frontend.

Open “Back End” codebase:

Open “repairrequest.routes.js”,

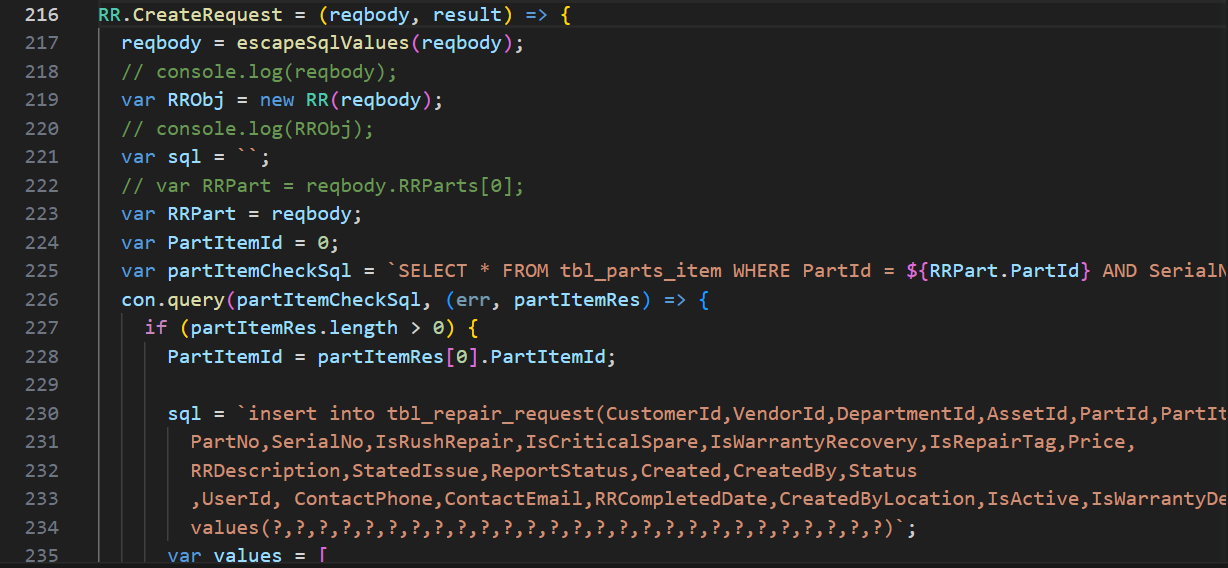


“Right” click on “RR.create” in line 17, click “Go to Definition”, it shows the code in “repairrequest.controller.js”,



and so on.

“Right” click on “CreateRequest” in line 58, click “Go to Definition”, it shows its “SQL” codebase in “repairrequest.model.js”,

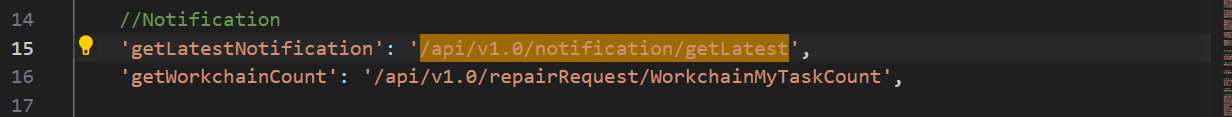


and so on.

1. **“/api/v1.0/notification/getLatest”**

Open “Front End” codebase”:

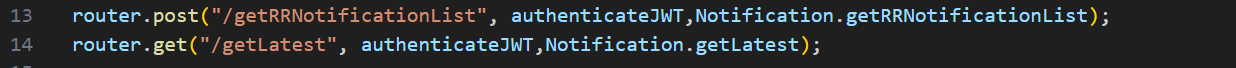
Search the “Customer Reference” API in “restURI.ts” file:



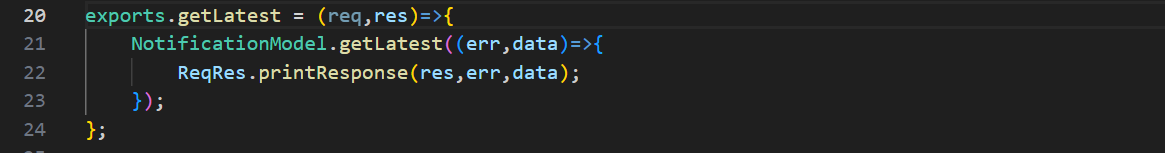
Copy: “getLatestNotification” to search in frontend.

Open “Back End” codebase:

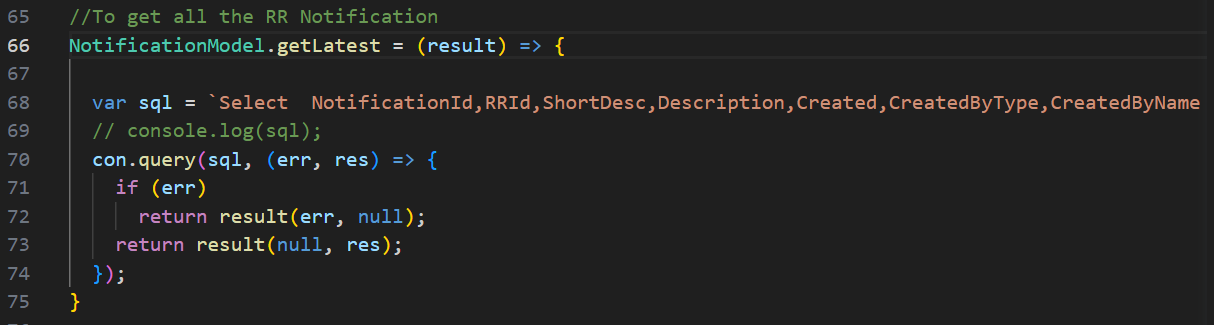
Open “notification.routes.js”,



“Right” click on “Notification.getLatest” in line 14, click “Go to Definition”, it shows the code in “notification.controller.js”,



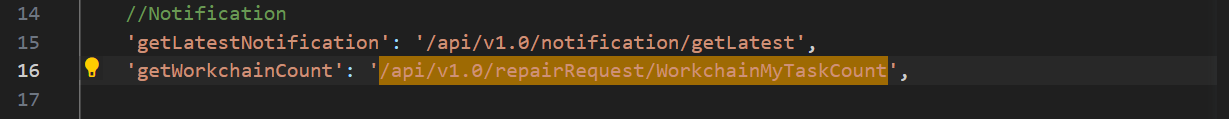
“Right” click on “getLatest” in line 21, click “Go to Definition”, it shows its “SQL” codebase in “notification.model.js”,



1. **“/api/v1.0/repairRequest/WorkchainMyTaskCount”**

Open “Front End” codebase”:

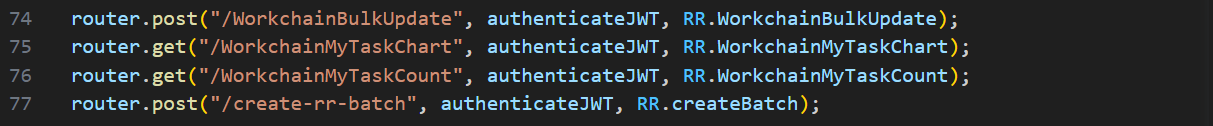
Search the “Customer Reference” API in “restURI.ts” file:



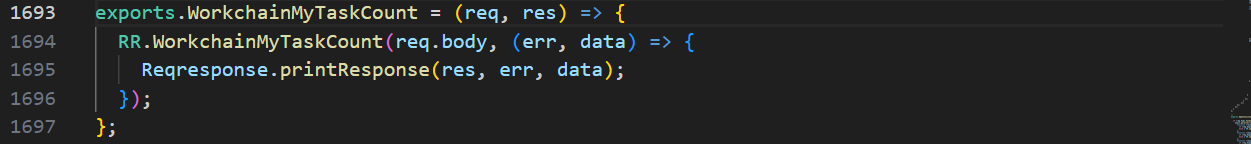
Copy: “getWorkchainCount” to search in frontend.

Open “Back End” codebase:

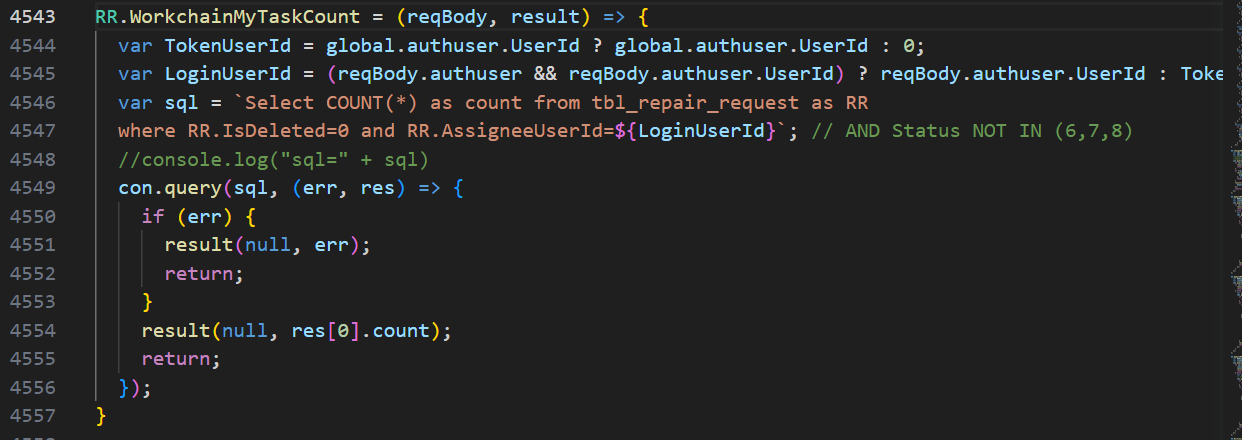
Open “repairrequest.routes.js”,



“Right” click on “RR.WorkchainMyTaskCount” in line 76, click “Go to Definition”, it shows the code in “repairrequest.controller.js”,



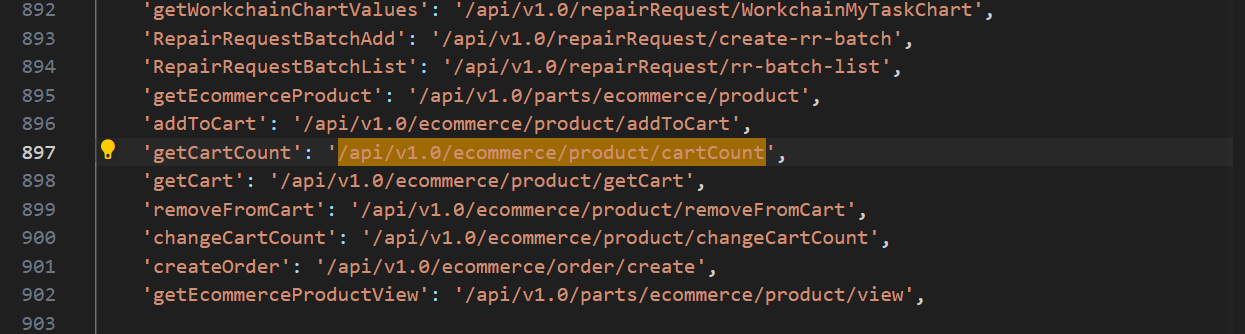
“Right” click on “WorkchainMyTaskCount” in line 1693, click “Go to Definition”, it shows its “SQL” codebase in “repairrequest.model.js”,



1. **“/api/v1.0/ecommerce/product/cartCount”**

Open “Front End” codebase”:

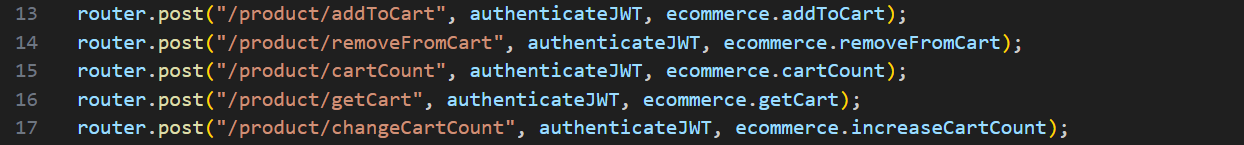
Search the “Customer Reference” API in “restURI.ts” file:



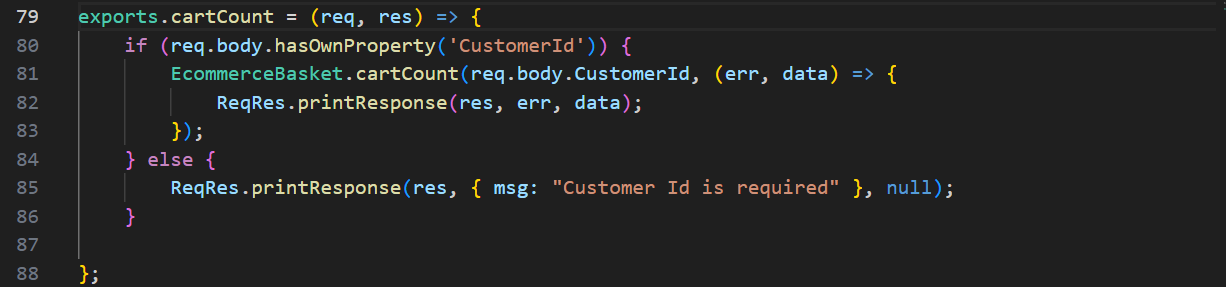
Copy: “getCartCount” to search in frontend.

Open “Back End” codebase:

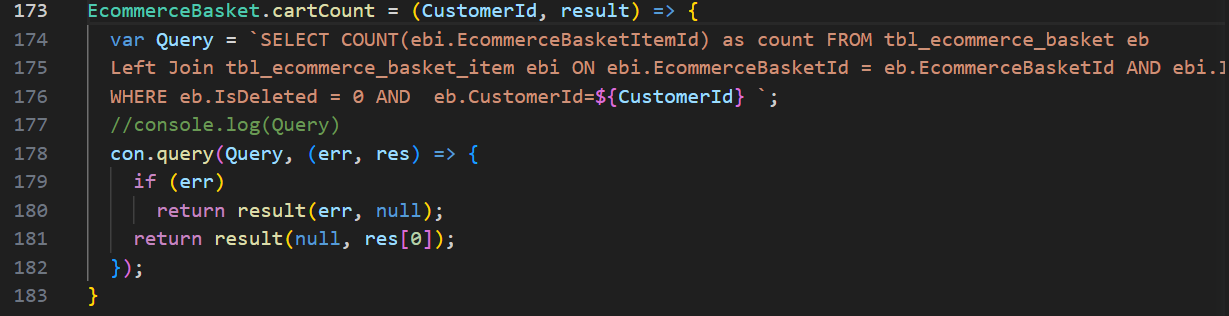
Open “ecommerce.routes.js”,



“Right” click on “ecommerce.cartCount” in line 15, click “Go to Definition”, it shows the code in “ecommerce.controller.js”,



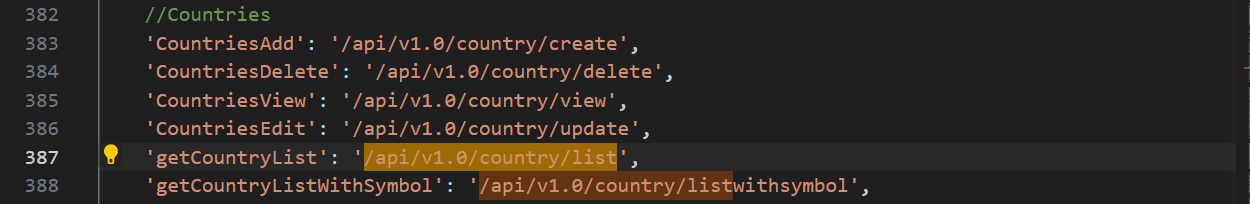
“Right” click on “cartCount” in line 81, click “Go to Definition”, it shows its “SQL” codebase in “ecommerce.basket.model.js”,



1. **“/api/v1.0/country/list”**

Open “Front End” codebase”:

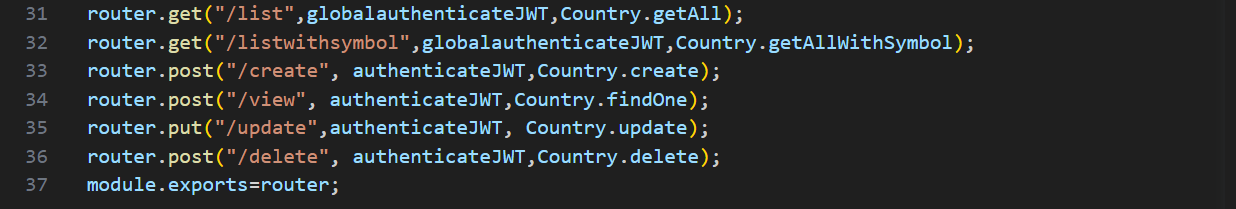
Search the “Customer Reference” API in “restURI.ts” file:



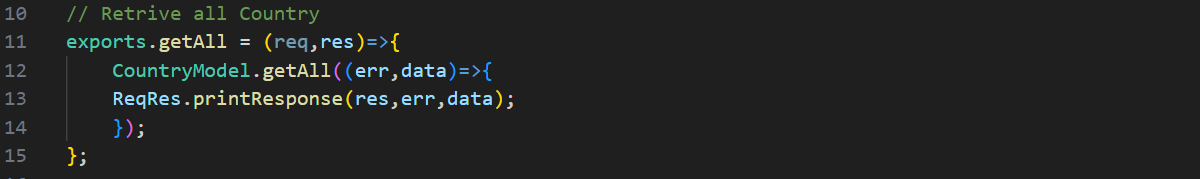
Copy: “getCountryList” to search in frontend.

Open “Back End” codebase:

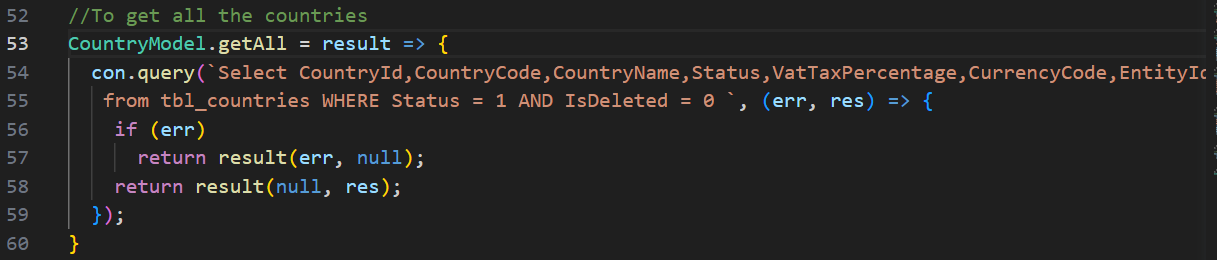
Open “country.routes.js”,



“Right” click on “Country.getAll” in line 31, click “Go to Definition”, it shows the code in “country.controller.js”,



“Right” click on “getAll” in line 11, click “Go to Definition”, it shows its “SQL” codebase in “country.model.js”,



1. **“/api/v1.0/repairRequest/view”**

Open “Front End” codebase”:

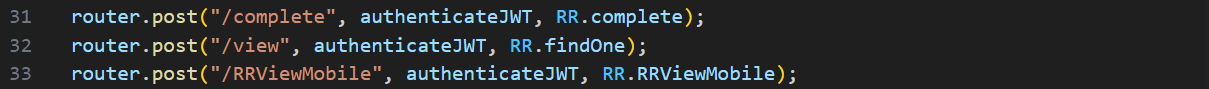
Search the “Customer Reference” API in “restURI.ts” file:



Copy: “RepairRequestView” to search in frontend.

Open “Back End” codebase:

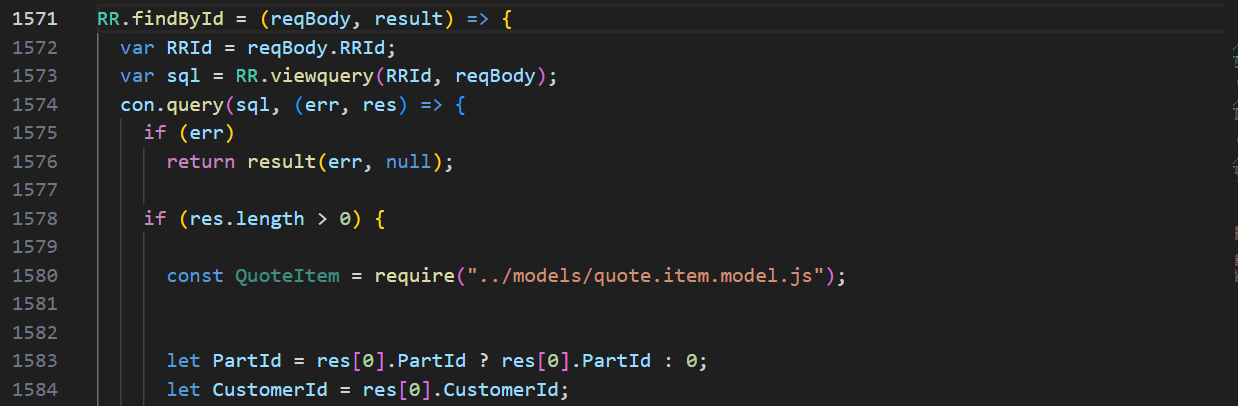
Open “repairrequest.routes.js”,



“Right” click on “RR.findOne” in line 32, click “Go to Definition”, it shows the code in “repairrequest.controller.js”,



“Right” click on “findById” in line 991, click “Go to Definition”, it shows its “SQL” codebase in “repairrequest.model.js”,



and so on.

1. **“/users/getAllActiveAdmin”**

Open “Front End” codebase”:

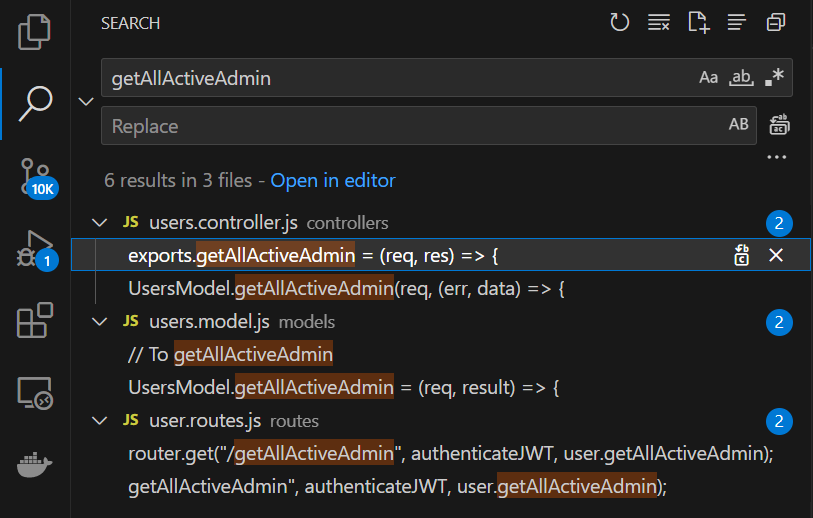
Search the “Customer Reference” API in “restURI.ts” file:



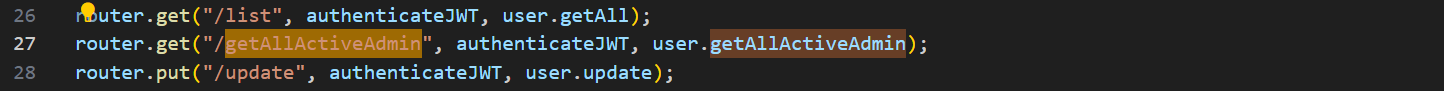
Copy: “getAllActiveAdmin” to search in frontend.

Open “Back End” codebase:

Search “getAllActiveAdmin” in the “Search” bar,



Click the “users.routes.js” file,



“Right” click “getAllActiveAdmin” in line 27, click “Go to Definition”, it goes to “users.controller.js.

Line 181 shows its code.

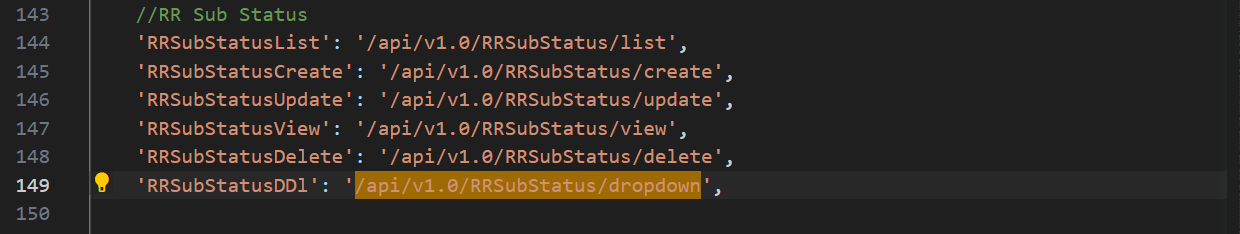
“Right” click “getAllActiveAdmin” in line 182, click “Go to Definition”, it goes to “users.model.js” and shows its “SQL” codebase in line 675.



1. **“/api/v1.0/RRSubStatus/dropdown”**

Open “Front End” codebase”:

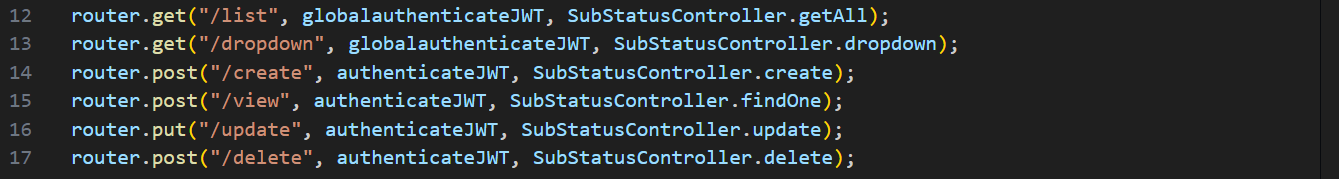
Search the “Customer Reference” API in “restURI.ts” file:



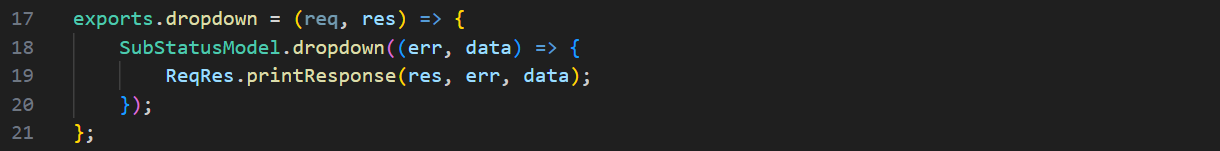
Copy: “RRSubStatusDDl” to search in frontend.

Open “Back End” codebase:

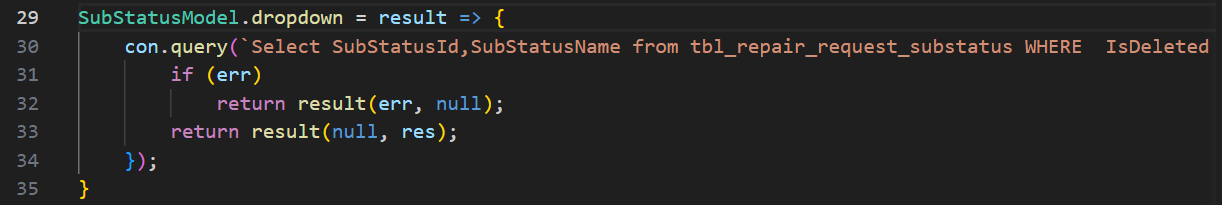
Open “substatus.routes.js”,



“Right” click on “SubStatusController.dropdown” in line 13, click “Go to Definition”, it shows the code in “substatus.controller.js”,



“Right” click on “dropdown” in line 18, click “Go to Definition”, it shows its “SQL” codebase in “substatus.model.js”,



1. **“/api/v1.0/currencyExchangeRate/exchange”**

Open “Front End” codebase”:

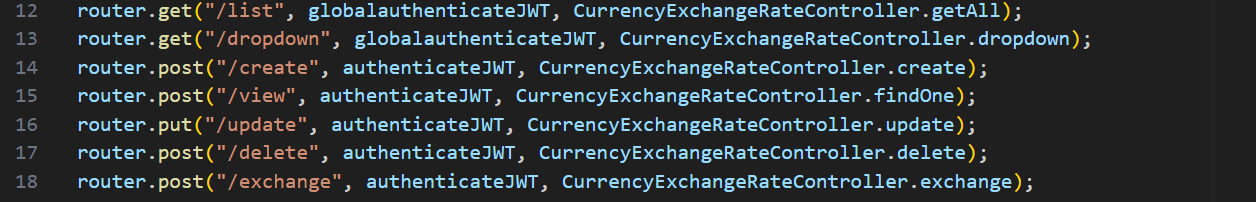
Search the “Customer Reference” API in “restURI.ts” file:



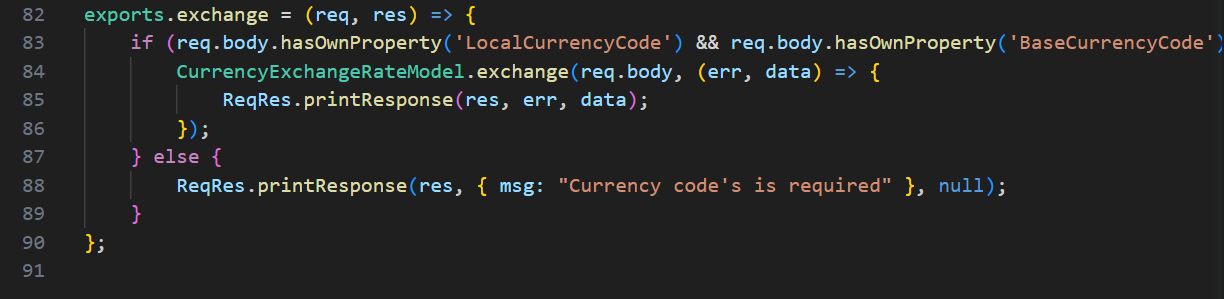
Copy: “Exchange” to search in frontend.

Open “Back End” codebase:

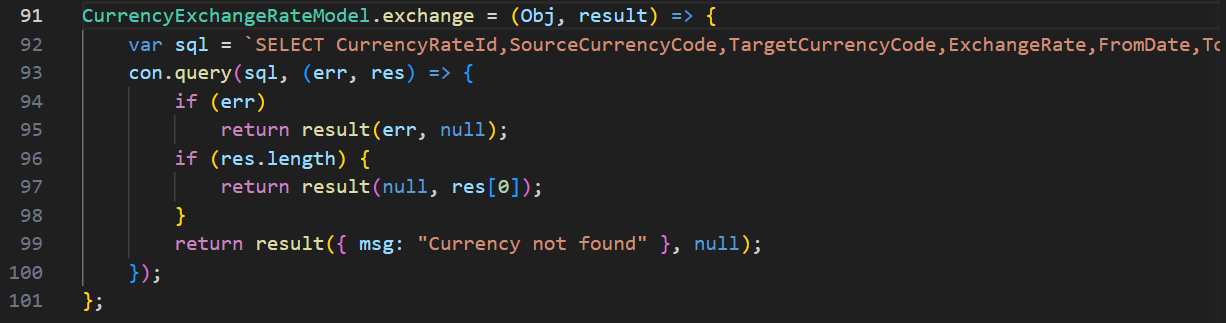
Open “currency.exchange.rate.routes.js”,



“Right” click on “CurrencyExchangeRateController.exchange” in line 18, click “Go to Definition”, it shows the code in “currency.exchange.rate.controller.js”,



“Right” click on “exchange” in line 84, click “Go to Definition”, it shows its “SQL” codebase in “currency.exchange.rate.model.js”,



1. **“/api/v1.0/customerParts/list”**: Repeated.