NewgenONE

Rule Builder Microservices

Developer Guide

Version: 2024.2

Disclaimer

This document contains information proprietary to Newgen Software Technologies Ltd. User may not disclose or use any proprietary information or use any part of this document without written permission from Newgen Software Technologies Ltd.

Newgen Software Technologies Ltd. makes no representations or warranties regarding any software or to the contents or use of this guide. It also specifically disclaims any express or implied warranties of merchantability, title, or fitness for any particular purpose. Even though Newgen Software Technologies Ltd. has tested the hardware and software and reviewed the documentation, it does not guarantee or imply that this document is error free or accurate regarding any particular specification. As a result, this product is sold as it is and user, the purchaser, is assuming the entire risk as to its quality and performance. Further, Newgen Software Technologies Ltd. reserves the right to revise this publication and make changes in its content without any obligation to notify any person, of such revisions or changes. Newgen Software Technologies Ltd. authorizes no Newgen agent, dealer or employee to make any modification, extension, or addition to the above statements.

Newgen Software Technologies Ltd. has attempted to supply trademark information about company names, products, and services mentioned in this document. Trademarks indicated below were derived from various sources.

Newgen Software, Registered Office, New Delhi
E-44/13
Okhla Phase - II
New Delhi 110020
India
Phone: +91 1146 533 200
info@newgensoft.com

Contents

Preface **5**
Revision history 5
About this guide 5
Intended audience 6
Related documents 6
Documentation feedback 6
Introduction to NewgenONE Rule Builder Microservices **7**
Prerequisites **7**
Rule endpoint fetching service **8**
getListrulePackages 8
Header parameters description 9
Request body parameters description 9
Query parameters description 9
Response body parameters description 10
Error codes and descriptions 11
Sample request and response body 12
getListruleFlows 13
Header parameters description 14
Request body parameters description 14
Query parameters description 14
Response body parameters description 15
Error codes and descriptions 16
Sample request and response body 18
getList 18
Header parameters description 19
Request body parameters description 19
Query parameters description 19
Response body parameters description 20
Error codes and descriptions 21
Sample request and response body 23
getInput 24
Header parameters description 25
Request body parameters description 25
Query parameters description 26
Response body parameters description 26
Error codes and descriptions 28
Sample request and response body 30
Rule execution service **32**
RuleExecutionService 32
Header parameters description 32

Request body parameters description 33
Query parameters description 38
Response body parameters description 39
Error codes and descriptions 39
Sample request and response body 42
Sample code 44

Preface

This chapter provides information about the purpose of this guide, details of the intended audience, revision history, and related documentation for NewgenONE Rule Builder Microservices.

Revision history

Revision date	Description
October 2024	Initial publication

About this guide

This developer guide describes the Microservices for fetching details of ruleFlows and rulePackages and their execution that are used in the Rule Builder.

- ruleFlow — Enables you to set the sequence for rule execution, either conditionally or unconditionally.
- rulePackage — Allows you to set a collection of business rules containing all the configurations to determine conditions and set actions for the designed rules.

> To ensure you are referring to the latest and most recent revision of this guide, download it from one of the following locations:
> Newgen Internal Doc Portal, if you are a Newgen employee.
> Newgen Partner Portal, if you are Newgen partner.

Intended audience

This guide is intended for the developers from internal or external product implementation teams responsible for integrating Application Programming Interface (APIs) exposed as NewgenONE Rule Builder Microservices with Custom Applications. The reader must be comfortable with API Signature formats in JSON to understand API requests and responses. The reader must have execution rights for the specific rulePackages and ruleFlows.

Related documents

The following documents are related to NewgenONE Rule Builder Microservices:

- NewgenONE 2024.2 Overview Guide
- NewgenONE 2024.2 Rule Builder User Guide
- NewgenONE 2024.2 Configuration and Deployment Guide for AWS
- NewgenONE 2024.2 Newgen Enterprise Product Containerization Guide for AWS
- NewgenONE 2024.2Developer Guide

Documentation feedback

To provide feedback or any improvement suggestions on technical documentation, write an email to docs.feedback@newgensoft.com.

To help capture your feedback effectively, share the following information in your email:

- Document name
- Version
- Chapter, topic, or section
- Feedback or suggestions

Introduction to NewgenONE Rule Builder Microservices

The NewgenONE Rule Builder Microservices consist of the exposed APIs that allow you to perform the following operations:

- Fetch the list of rulePackage and ruleFlow.
- Fetch the input JSON required for executing a particular ruleFlow or rulePackage.
- Execute the rulePackage and ruleFlow.

Ensure that the signed-in user must have execution rights for the rulePackage and ruleFlow execution. Also, users can view the list whose package rights are assigned to them.

Prerequisites

The prerequisites are as follows:

- For deployment of Rule Builder as Microservices, update the **IsMicroservice** tag value from *<IsMicroservice>N</IsMicroservice>* to *<IsMicroservice>Y</IsMicroservice>*in the *pm_webappcontext.xml* file.
- For deployment of Rule Builder as Microservices, update the **IsMicroserviceEnabled=Y** in the in file *bin\Newgen\NGConfig\omniflowconfiguration\brwebconfig\brconfig.ini*.

Rule endpoint fetching service

This chapter includes the following APIs:

- getListrulePackages
- getListruleFlows
- getList
- getInput

getListrulePackages

The GetlistrulePackages API fetches the list of rulePackages deployed as a rest service and the signed-in user can execute them. For more details, you can refer to the JSON request and response parameters provided in the next sub-sections.

- URL — *http://<GatewayIp:GatewayPort >/rule-endpoints-fetching-service-context/brms/getList/rulePackages*
  Where, GatewayIp is the IP of the machine, and GatewayPort is the port of the machine where the rule endpoint fetching service is deployed.
- Request Media Type — application/json
- Response Media Type — application/json
- Method Type — POST

Header parameters description

The following table describes the header parameters of the *GetlistRulePackages* API if OAuth is disabled:

Name	Parameter Scope	Type	Description	Mandatory
Authorization	HeaderParam	string	The sessionID of the user.	Yes
Tenant-Id	HeaderParam	string	Name of the Cabinet.	Yes
Accept-Language	HeaderParam	string	Name of the Locale.	No

If OAuth is enabled:

Name	Parameter Scope	Type	Description	Mandatory
AppId	HeaderParam	String	AppId of the cabinet associated	Yes
SessionValid	HeaderParam	String	SessionValid Token received from the OD Call.	Yes
AccessToken	HeaderParam	String	AccessToken received from the OD Call.	Yes

In case the user is part of the immune group then sessionID is passed as an authorization header parameter.

Request body parameters description

The *GetlistRulePackages* API does not include the request body parameters.

Query parameters description

The *GetlistRulePackages* API does not include the query parameters.

Response body parameters description

The following table describes the response body parameters of the *GetlistRulePackages* API:

Name	Parent Tag	Type	Description
rulePackages	Root	Array	Contains the list of all rulePackages on which the user has execution rights. • If the user has no rights on any rulePackages or there are no rulePackages in the system, then the response is as an empty array.
versionNo	rulePackages	Integer	Displays the version number of the rulePackage.
ruleParentId	rulePackages	Integer	Displays the ID of the rulePackage.
displayMode	rulePackages	String	Represents N as the Normal display mode of rulePackage.
ruleParentType	rulePackages	String	Represents P as rulePackage which is a type of the ruleParent.
encrypted	rulePackages	String	Represents Y for an encrypted rulePackage, while N represents a non-encrypted rulePackage.
cabinetName	rulePackages	String	Contains the cabinet on which the RulePackage exists.
name	rulePackages	String	Displays the name of the RulePackage.

Error codes and descriptions

The following table describes the error codes that can occur while executing the *GetlistRulePackages* API:

Error Code	HTTP Status	HTTP Status Description	Error Code Description	Description
N/A	401	Unauthorized	Invalid Session	Regenerate the session and pass the valid session ID in the authorization header.
BPM_BRM_003	415	Unsupported Media Type	HTTP Method is invalid	Change the HTTP method to GET.
BPM_BRM_006	400	Bad Request	Required Request Header Missing	Enter all the required headers.
BPM_BRM_007	500	Internal Server Error	SQL Exception	Contact your Newgen representative.
BPM_BRM_008	400	Bad Request	Operation Failed	Check your request or contact support
BPM_BRM_008	500	Internal Server Error	Operation Failed	Contact your Newgen representative.
BPM_BRM_009	403	Forbidden	User has no rights on rulePackage/ ruleFlow	Business admin must provide the execution rights.
BPM_BRM_011	405	Method Not Allowed	Method not allowed for this cabinet profile (ibps5) at Fetching Service	Enter the correct endpoint for the cabinet profile. For example, if you have entered root instead of a string so it indicates the Method is not allowed error.

Error Code	HTTP Status	HTTP Status Description	Error Code Description	Description
BPM_BRM_012	400	Bad Request	Invalid input parameters at Fetching Service	Enter the valid input parameters.
BPM_BRM_015	503	Service Unavailable	Service is either unavailable or slow	Service is either unavailable or slow.
BPM_BRM_016	401	Unauthorized	Session is invalid, please re-authenticate	Sign in again and use the new sessionID.
BPM_BRM_017	503	Service Unavailable	Redis server is unavailable	Contact your Newgen representative.
BPM_BRM_018	503	Service Unavailable	Fetching Service is either unavailable or slow	Service is either unavailable or slow.
BPM_BRM_019	405	Conflict	RulePackage/Flow does not exist	Create this rulePackage or ruleFlow on the cabinet used during sign in.
BPM_BRM_020	409	Conflict	Data source is not valid	Check your database entries to ensure correct cabinet details.

Sample request and response body

Sample Response:


```
{
  "rulePackages": [
    {
      "versionNo": 1,
      "ruleParentId": 1,
      "displayMode": "N",
      "ruleParentType": "P",
      "encrypted": "N",
```

```
      "name": "HomeLoan_RulePackage"
    },
    {
      "versionNo": 1,
      "ruleParentId": 5,
      "displayMode": "N",
      "ruleParentType": "P",
      "encrypted": "N",
      "name": "EmployeePerformanceScore_RulePackage"
    },
    {
      "versionNo": 1,
      "ruleParentId": 6,
      "displayMode": "N",
      "ruleParentType": "P",
      "encrypted": "N",
      "name": "carLoan_RulePackage"
    },
    {
      "versionNo": 1,
      "ruleParentId": 7,
      "displayMode": "N",
      "ruleParentType": "P",
      "encrypted": "N",
      "name": "CIBILCalculator_RulePackage"
    }
  ]
}
```


getListruleFlows

The Getlistruleflows API fetches the list of ruleFlows deployed as a rest service and the signed-in user can execute them. For more details, you can refer to the JSON request and response parameters provided in the next sub-sections.

- URL — *http://<GatewayIp:GatewayPort >/rule-endpoints-fetching-service-context/brms/getList/ruleFlows*
  Where, GatewayIp is the IP of the machine, and GatewayPort is the port of the machine where the rule endpoints fetching service is deployed.
- Request Media Type — application/json
- Response Media Type — application/json
- Method Type — POST

Header parameters description

The following table describes the header parameters of the *Getlistruleflows* API if OAuth is disabled:

Name	Parameter Scope	Type	Description	Mandatory
Authorization	HeaderParam	string	The sessionID of the user.	Yes
Tenant-Id	HeaderParam	string	Name of the Cabinet.	Yes
Accept-Language	HeaderParam	string	Name of the Locale.	No

If OAuth is enabled:

Name	Parameter Scope	Type	Description	Mandatory
AppId	HeaderParam	String	AppId of the cabinet associated	Yes
SessionValid	HeaderParam	String	SessionValid Token received from the OD Call.	Yes
AccessToken	HeaderParam	String	AccessToken received from the OD Call.	Yes

In case the user is part of the immune group then sessionID is passed as an authorization header parameter.

Request body parameters description

The *Getlistruleflows* API does not include the request body parameters.

Query parameters description

The *Getlistruleflows* API does not include the query parameters.

Response body parameters description

The following table describes the response body parameters of the *GetlistRuleflows* API:

Name	Parent Tag	Type	Description
ruleFlows	Root	Array	Contains the List of all ruleFlows. If the user has no rights on any ruleFlows or there are no ruleFlows in the system, then the response is as an empty array.
versionNo	ruleFlows	Integer	Displays the version number of the ruleFlow.
ruleParentId	ruleFlows	Integer	Contains the ruleFlow ID.
displayMode	ruleFlows	String	Displays the normal display mode of ruleFlow, that is, N.
ruleParentType	ruleFlows	String	Represents P as rulePackage which is a type of ruleFlow.
encrypted	ruleFlows	String	Represents Y for encrypted ruleFlow, while N represents a non-encrypted ruleFlow.
cabinetName	ruleFlows	String	Contains the cabinet on which the ruleFlow exists.
name	ruleFlows	String	Displays the name of the ruleFlow.

Error codes and descriptions

The following table describes the error codes that can occur while executing the *GetlistRuleflows* API:

Error Code	HTTP Status	HTTP Status Description	Error Code Description	Comments
N/A	401	Unauthorized	Invalid Session	Regenerate the session and pass the valid session ID in the authorization header.
BPM_BRM_003	415	Unsupported Media Type	HTTP Method is invalid	Change the HTTP method to GET.
BPM_BRM_006	400	Bad Request	Required Request Header Missing	Enter all the required headers.
BPM_BRM_007	500	Internal Server Error	SQL Exception	Contact your Newgen representative.
BPM_BRM_008	400	Bad Request	Operation Failed	Check your request or contact support.
BPM_BRM_008	500	Internal Server Error	Operation Failed	Contact your Newgen representative.
BPM_BRM_009	403	Forbidden	User has no rights on rulePackage/ Flow	Business admin must provide execution the rights.

Error Code	HTTP Status	HTTP Status Description	Error Code Description	Comments
BPM_BRM_011	405	Method Not Allowed	Method not allowed for this cabinet profile(ibps5) at Fetching Service	Enter the correct endpoint for the cabinet profile. For example, if you have entered root instead of a string then it indicates the error that Method is not allowed.
BPM_BRM_012	400	Bad Request	Invalid input parameters at Fetching Service	Enter the valid input parameters.
BPM_BRM_015	503	Service Unavailable	Service is either unavailable or slow	Service is either unavailable or slow.
BPM_BRM_016	401	Unauthorized	Session is invalid, re-authenticate	Sign in again and use the new session ID.
BPM_BRM_017	503	Service Unavailable	Redis server is unavailable	Contact your Newgen representative.
BPM_BRM_018	503	Service Unavailable	Fetching Service is either unavailable or slow	Service is either unavailable or slow.
BPM_BRM_019	405	Conflict	rulePackage/Flow does not exist	Create this rulePackage or ruleFlow on the cabinet used during sign in.
BPM_BRM_020	409	Conflict	Data source is not valid	Check your database entries to ensure correct cabinet details.

Sample request and response body

Sample Response:

```
{
  "ruleFlows": [
    {
      "versionNo": 1,
      "ruleParentId": 1,
      "displayMode": "N",
      "ruleParentType": "F",
      "encrypted": "N",
      "name": "PPF_RuleFlow"
    },
    {
      "versionNo": 1,
      "ruleParentId": 3,
      "displayMode": "N",
      "ruleParentType": "F",
      "encrypted": "N",
      "name": "Sale_RuleFlow"
    }
  ]
}
```

getList

The Getlist API fetches the list of ruleFlows and rulePackages deployed as a rest service and the signed-in user can execute them. For more details, you can refer to the JSON request and response parameters provided in the next sub-sections.

- URL — *http://<GatewayIp:GatewayPort >/rule-endpoints-fetching-service-context/brms/getList*
  Where, GatewayIp is the IP of the machine, and GatewayPort is the port of the machine where the rule endpoints fetching service is deployed.
- Request Media Type — application/json
- Response Media Type — application/json
- Method Type — POST

Header parameters description

The following table describes the header parameters of the *Getlist* API if OAuth is disabled:

Name	Parameter Scope	Type	Description	Mandatory
Authorization	HeaderParam	string	The sessionID of the user	Yes
Tenant-Id	HeaderParam	string	Name of the Cabinet.	Yes
Accept-Language	HeaderParam	string	Name of the Locale.	No

If OAuth is enabled:

Name	Parameter Scope	Type	Description	Mandatory
AppId	HeaderParam	String	AppId of the cabinet associated	Yes
SessionValid	HeaderParam	String	SessionValid Token received from the OD Call.	Yes
AccessToken	HeaderParam	String	AccessToken received from the OD Call.	Yes

In case the user is part of the immune group then sessionID is passed as an authorization header parameter.

Request body parameters description

The Getlist API does not include the request body parameters.

Query parameters description

The *Getlist* API does not include the query parameters.

Response body parameters description

The following table describes the response body parameters of the *Getlist* API:

Name	Parent Tag	Type	Description
rulePackages	Root	Array	Contains the all rulePackages list on which the user has execution rights. • If the user has no rights on any rulePackages or there are no rulePackages in the system, then the response is as an empty array.
ruleFlows	Root	Array	Contains the ruleFlows list on which the user has execution rights. • If the user has no rights on any ruleFlows or there are no ruleFlows in the system, then the response is as an empty array.
versionNo	rulePackages	Integer	Indicates the version number of the rulePackage.
ruleParentId	rulePackages	Integer	Displays the rulePackage ID.
displayMode	rulePackages	String	Displays the normal display mode or rulePackage, that is, N.
ruleParentType	rulePackages	String	Represents P as rulePackage which is a type of the ruleParent.
encrypted	rulePackages	String	Displays Y for an encrypted rulePackage, while N displays a non-encrypted rulePackage.
cabinetName	rulePackages	String	Contains the cabinet on which the rulePackage exists.
name	rulePackages	String	Displays the name of the rulePackage.

Name	Parent Tag	Type	Description
versionNo	ruleFlows	Integer	Contains the version number of the ruleFlow.
ruleParentId	ruleFlows	Integer	Displays the ruleFlow ID.
displayMode	ruleFlows	String	Displays N as the Normal display mode of ruleFlow.
ruleParentType	ruleFlows	String	Represents P as rulePackage which is a type of ruleFlow.
encrypted	ruleFlows	String	Represents Y as an encrypted ruleFlow, while N represents a non-encrypted ruleFlow.
cabinetName	ruleFlows	String	Contains the cabinet on which the ruleFlow exists.
name	ruleFlows	String	Displays the name of the ruleFlow.

Error codes and descriptions

The following table describes the error codes that can occur while executing the *Getlist* API:

Error Code	HTTP Status	HTTP Status Description	Error Code Description	Comments
N/A	401	Unauthorized	Invalid Session	Regenerate the session and pass the valid session ID in the authorization header.
BPM_BRM_003	415	Unsupported Media Type	HTTP Method is invalid	Change the HTTP method to GET.
BPM_BRM_006	400	Bad Request	Required Request Header Missing	Enter all the required headers.

Error Code	HTTP Status	HTTP Status Description	Error Code Description	Comments
BPM_BRM_007	500	Internal Server Error	SQL Exception	Contact your Newgen representative.
BPM_BRM_008	400	Bad Request	Operation Failed	Check your request or contact support.
BPM_BRM_008	500	Internal Server Error	Operation Failed	Contact your Newgen representative.
BPM_BRM_009	403	Forbidden	User has no rights on Rule Package/ Flow	Business admin must provide the execution rights.
BPM_BRM_011	405	Method Not Allowed	Method not allowed for this cabinet profile(ibps5) at Fetching Service	Enter the correct endpoint for the cabinet profile. For example, if you have entered root instead of a string then it indicates "Method is not allowed" error.
BPM_BRM_012	400	Bad Request	Invalid input parameters at Fetching Service	Enter the valid input parameters.
BPM_BRM_015	503	Service Unavailable	Service is either unavailable or slow	Service is either unavailable or slow.
BPM_BRM_016	401	Unauthorized	Session is invalid, please re-authenticate	Sign in again and use the new session ID.
BPM_BRM_017	503	Service Unavailable	Redis server is unavailable	Contact your Newgen representative.
BPM_BRM_018	503	Service Unavailable	Fetching Service is either unavailable or slow	Service is either unavailable or slow.

Error Code	HTTP Status	HTTP Status Description	Error Code Description	Comments
BPM_BRM_019	405	Conflict	rulePackage/Flow does not exist	Create this rulePackage or ruleFlow on the cabinet used during sign in.
BPM_BRM_020	409	Conflict	Data source is not valid	Check your database entries to ensure correct cabinet details.

Sample request and response body

Sample Response:


```
{
  "rulePackages": [
    {
      "versionNo": 1,
      "ruleParentId": 1,
      "displayMode": "N",
      "ruleParentType": "P",
      "encrypted": "N",
      "name": "HomeLoan_RulePackage"
    },
    {
      "versionNo": 1,
      "ruleParentId": 5,
      "displayMode": "N",
      "ruleParentType": "P",
      "encrypted": "N",
      "name": "EmployeePerformaceScore_RulePackage"
    },
    {
      "versionNo": 1,
      "ruleParentId": 6,
      "displayMode": "N",
      "ruleParentType": "P",
      "encrypted": "N",
      "name": "CarLoan_RulePackage"
    },
    {
      "versionNo": 1,
```

```
      "ruleParentId": 7,
      "displayMode": "N",
      "ruleParentType": "P",
      "encrypted": "N",
      "name": "CIBILCalculator_RulePackage"
    }
  ],
  "ruleFlows": [
    {
      "versionNo": 1,
      "ruleParentId": 1,
      "displayMode": "N",
      "ruleParentType": "F",
      "encrypted": "N",
      "name": "PPF_RuleFlow"
    },
    {
      "versionNo": 1,
      "ruleParentId": 3,
      "displayMode": "N",
      "ruleParentType": "F",
      "encrypted": "N",
      "name": "Sale_RuleFlow"
    }
  ]
}
```


getInput

The GetInput API fetches the input JSON that you can use as a body for the execution of /brms/exec of a particular ruleFlow or rulePackage. For more details, you can refer to the JSON request and response parameters provided in the next sub-sections.

- URL — *http://<GatewayIp:GatewayPort >/rule-endpoints-fetching-service-context/brms/getInput*
  Where, GatewayIp is the IP of the machine, and GatewayPort is the port of the machine where the rule endpoints fetching service is deployed.
- Request Media Type — application/json
- Response Media Type — application/json
- Method Type — POST

Header parameters description

The following table describes the header parameters of the *GetInput* API if OAuth is disabled:

Name	Parameter Scope	Type	Description	Mandatory
Authorization	HeaderParam	string	The sessionID of the user.	Yes
Tenant-Id	HeaderParam	string	Name of the Cabinet.	Yes
Accept-Language	HeaderParam	string	Name of the Locale.	No

If OAuth is enabled:

Name	Parameter Scope	Type	Description	Mandatory
AppId	HeaderParam	String	AppId of the cabinet associated	Yes
SessionValid	HeaderParam	String	SessionValid Token received from the OD Call.	Yes
AccessToken	HeaderParam	String	AccessToken received from the OD Call.	Yes

In case the user is part of the immune group then sessionID is passed as an authorization header parameter.

Request body parameters description

The following table describes the request body parameters of the *GetInput* API:

Name	Parameter Scope	Type	Description	Valid Values	Mandatory
ruleParentId	Body Parameter	integer	Indicates the ID of the rulePackage or ruleFlow that requires its JSON body and URL.	ruleParentId>=1	Yes

Name	Parameter Scope	Type	Description	Valid Values	Mandatory
ruleParentType	Body Parameter	string	Represents the value for rulePackage and ruleFlow execution.	P - For rulePackage execution F - For ruleFlow execution	Yes
versionNo	Body Parameter	float	Indicates the ID of the rulePackage or ruleFlow that requires its JSON body and URL.	versionNo = -1.0 OR versionNo >=1.0	Yes

Query parameters description

The *GetInput* API does not include the query parameters.

Response body parameters description

The following table describes the response body parameters of the *GetInput* API:

Name	Parent Tag	Type	Description
url	root	String	Specifies the endpoint containing the REST service for the requested ruleFlow or rulePackage.
jsonBody	root	Object	Contains the JSON which is the Request Body for the URL in the URL tag. Replace all the "?" symbols with your input values.
input	jsonBody	Array	Contains an array holding the input entity classes for rule execution.

Name	Parent Tag	Type	Description
className	input	String	Displays the name of the entity class.
classIsArray	input	String	Specifies if the entity class is an array entity class.
member	input	Array	Indicates the array that contains the entity members and their properties belonging to the parent entity class.
name	member	String	Displays the name of the entity member.
type	member	Integer	Indicates the data type of the entity member.
value	member	String	Indicates that you must replace '?' (default value of entity member) with a proper input value while inserting it into the JSON input body of Rule Builder or execution endpoint.
memberIsArray	member	String	Specifies whether the entity member is array type or not.
output	jsonBody	Array	Contains an array holding the output entity classes for rule execution.
className	output	String	Displays the name of the entity class.
classIsArray	output	string	Specifies if the entity class is an array entity class.
member	output	array	Contains an array of entity members and their properties belonging to the parent entity class.
name	member	string	Displays the name of the entity member.
type	member	integer	Indicates the data type of the entity member.

Name	Parent Tag	Type	Description
value	member	string	Contains the value of the entity member, this comes with a defined default value during the creation of the entity member from the Rule Builder interface.
memberIsArray	member	string	Indicates if the entity member is an array type or not.

Error codes and descriptions

The following table describes the error codes that can occur while executing the *GetInput* API:

Error Code	HTTP Status	HTTP Status Description	Error Code Description	Comments
N/A	401	Unauthorized	Invalid Session	Regenerate the session and pass the valid session ID in the authorization header.
BPM_BRM_003	415	Unsupported Media Type	HTTP Method is invalid	Change the HTTP method to GET.
BPM_BRM_006	400	Bad Request	Required Request Header Missing	Enter all the required headers.
BPM_BRM_007	500	Internal Server Error	SQL Exception	Contact your Newgen representative.
BPM_BRM_008	400	Bad Request	Operation Failed	Check your request or contact support.
BPM_BRM_008	500	Internal Server Error	Operation Failed	Contact your Newgen representative.
BPM_BRM_009	403	Forbidden	User has no rights on Rule Package/ Flow	Business admin must provide the execution rights.

Error Code	HTTP Status	HTTP Status Description	Error Code Description	Comments
BPM_BRM_011	405	Method Not Allowed	Method not allowed for this cabinet profile(ibps5) at Fetching Service	Enter the correct endpoint for the cabinet profile. For example, if you have entered root instead of a string so it indicates the Method is not allowed error .
BPM_BRM_012	400	Bad Request	Invalid input parameters at Fetching Service	Enter the valid input parameters.
BPM_BRM_015	503	Service Unavailable	Service is either unavailable or slow	Service is either unavailable or slow.
BPM_BRM_016	401	Unauthorized	Session is invalid, please re-authenticate	Sign in again and use the new session ID.
BPM_BRM_017	503	Service Unavailable	Redis server is unavailable	Contact your Newgen representative.
BPM_BRM_018	503	Service Unavailable	Fetching Service is either unavailable or slow	Service is either unavailable or slow.
BPM_BRM_019	405	Conflict	rulePackage/Flow does not exist	Create this rulePackage or ruleFlow on the cabinet used during sign in.
BPM_BRM_020	409	Conflict	Data source is not valid	Check your database entries to ensure correct cabinet details.

Sample request and response body

Sample Request:

```
{
    "ruleParentId" : 1,
    "ruleParentType" : "P",
    "versionNo" : 1.0
}
```

Sample Response:

```
{
  "url": "http://<IP>:<PORT>/rule-execution-service-context/brms/exec?
versionNo=1.0&rulesetName=emp_1_Test_RF&cabinetName=ng1cabinet&RuleFlow
=Y&encrypted=N&rulePackageID=27",
  "jsonBody": {
    "input": [
      {"className": "employeeDetails",
        "classIsArray": "N",
        "member": [
          {
            "memberIsArray": "N",

            "name": "employeeDetails",
            "type": 10,
            "value": "joe"
          }
        ]
      }
    ],
    "output": [
      {

        "className": "employeeDetails",
        "classIsArray": "N",
        "member": [
          {
            "memberIsArray": "N",

            "name": "employeeId",
            "type": 3,
            "value": "25"
          },
```

```
            {
              "memberIsArray": "Y",

              "name": "employeeRegions",
              "type": 10,
              "value": "East,South"
            }
          ]
        }
      ]
  }
}
```

Rule execution service

This chapter includes the RuleExecutionService API.

RuleExecutionService

The RuleExecutionService API allows you to execute the rulePackage or ruleFlow that is exposed as REST services. For more details, refer to the JSON request and response parameters provided in the next sub-sections.

- URL — *http://<GatewayIp:GatewayPort >/rule-execution-service-context/brms/exec*
  Where, *GatewayIp* and *GatewayPort* are respectively the IP and Port of the machine where the rule execution service is deployed.
- Request Media Type — application/json
- Response Media Type — application/json
- Method Type — POST

In the case of Docker and JWS server, if you execute an external function from Test via REST, the external function file must be present in the JWS server; otherwise, it will throw an error.

Header parameters description

The following table describes the header parameters of the *RuleExecutionService* API if OAuth is disabled:

Parameter	Parent Tag	Type	Description	Mandatory
Accept-Language	HeaderParam	Locale	Represents the specific region or language. Example: en-US	No
Tenant-Id	HeaderParam	String	Name of the cabinet.	Yes

Parameter	Parent Tag	Type	Description	Mandatory
User-Id	HeaderParam	String	Represents the user ID of the signed-in user ID.	No
Authorization	HeaderParam	String	Session ID to validate the session.	Yes
Token-Id	HeaderParam	String	A user password for two-way authentication.	No
UserName	HeaderParam	String	Represents the signed-in user name.	No

If OAuth is enabled:

Name	Parameter Scope	Type	Description	Mandatory
AppId	HeaderParam	String	AppId of the cabinet associated	Yes
SessionValid	HeaderParam	String	SessionValid Token received from the OD Call.	Yes
AccessToken	HeaderParam	String	AccessToken received from the OD Call.	Yes

In case the user is part of the immune group then sessionID is passed as an authorization header parameter.

Request body parameters description

The following table describes the request body parameters of the *RuleExecutionService* API:

Name	Parent Tag	Type	Description	Valid Values	Default Values	Mandatory
input	Root	Array of entity class objects	Classes are used as input in the rulePackage or ruleFlow.	Not Applicable (NA)	NA	Yes
className	Input	String	The utilized input class name.		NA	Yes

Name	Parent Tag	Type	Description	Valid Values	Default Values	Mandatory
classIsArray	Input	String	Indicates whether the used input class is an array or not.	Y- Yes N - No	NA	Yes
member	Input	Array member info	Represents the member used in input class has an array object.		NA	Yes
memberIsArray	member	String	Indicates whether the member is an array type or not.		NA	Yes
name	member	String	Name of the entity member.		NA	Yes

Name	Parent Tag	Type	Description	Valid Values	Default Values	Mandatory
type	member	Integer	The data type of the entity member.	3 — Integer 4 — Long 6 — Float 8 — Date 10 — String 12 — Boolean 13 — Integer Array 14 — Long Array 16 — Float Array 20 — String Array 22 — Boolean Array 30 — BigDecimal 40 — BigDecimal Array	NA	Yes
value	member	String	Indicates the client must provide the entity member value in the rule input, replacing the '?'.		NA	Yes
output	Root	Array of Entity class objects	Classes are used as output in the rulePackage or ruleFlow.	NA	NA	Yes
className	output	String	Name of the output class used in rule actions.		NA	Yes

Name	Parent Tag	Type	Description	Valid Values	Default Values	Mandatory
classIsArray	output	String	Determines whether the output class used is an array or not.	Y - Yes N - No	NA	Yes
member	output	Array of member info	Indicates an array object of members used in the output class.		NA	Yes
memberIsArray	member	String	Indicates whether the member is an array type or not.	Y - Yes N - No	NA	Yes
name	member	String	The name of the entity member.		NA	Yes

Name	Parent Tag	Type	Description	Valid Values	Default Values	Mandatory
type	member	Integer	The data type of the entity member.	3 — Integer 4 — Long 6 — Float 8 — Date 10 — String 12 — Boolean 13 — Integer Array 14 — Long Array 16 — Float Array 18 — Date Array 20 — String Array 22 — Boolean Array 30 — BigDecimal 40 — BigDecimal Array	NA	Yes
value	member	String	The default value of entity member.	Default value of the entity member, set by the user during deployment.	NA	Yes

Query parameters description

The following table describes the query parameters of the *RuleExecutionService* API:

Name	Parameter Scope	Type	Description	Valid Values	Mandatory
versionNo	Request Param	String	The version number of the rulePackage or ruleFlow.	-1 for version free execution 1≥ for floating	Yes
rulesetName	Request Param	String	The name of the rulePackage or ruleFlow.		Yes
isRuleFlow	Request Param	String	Represents whether the execution is for rulePackage or ruleFlow.	Y - Execution for rulePackage N - Execution for ruleFlow	Yes
encrypted	Request Param	String	Indicates whether the ruleFlow or rulePackage is encrypted or not.	Y - Yes N - No	Yes
rulePackageID	Request Param	String	The ID of ruleFlow or rulePackage.	Positive integer value	Yes

Response body parameters description

The following table describes the response body parameters of the *RuleExecutionService* API:

Name	Parent Tag	Parameter Type	Description
rulesExecuted	Root	Array of String	Executed rules with comma-separated string.
status	Root	String	Upon successful execution of the rulePackage, success returns, otherwise, failure is the result.
output	Root	Output classes objects	The output value of classes along with their member.

Error codes and descriptions

The following table describes the error codes that can occur while executing the *RuleExecutionService* API:

Error Code	HTTP Status	HTTP Status Description	Error Code Description	Comments
N/A	401	Unauthorized	Invalid Session	Regenerate the session and pass the valid session ID in the authorization header.
BPM_BRM_003	415	Unsupported Media Type	HTTP Method is invalid	Change the HTTP method to POST.
BPM_BRM_006	400	Bad Request	Required Request Header Missing	Enter all the required headers.

Error Code	HTTP Status	HTTP Status Description	Error Code Description	Comments
BPM_BRM_007	500	Internal Server Error	SQL Exception	Contact your Newgen representative.
BPM_BRM_008	400	Bad Request	Operation Failed	Check your request or contact support.
BPM_BRM_008	500	Internal Server Error	Operation Failed	Contact your Newgen representative.
BPM_BRM_014	400	Bad Request	Invalid RulePackage/ Ruleflow ID in URL	Enter the valid RulePackage or RuleFlow ID.
BPM_BRM_015	400	Bad Request	Invalid RulePackage or RuleFlow Name in URL	Enter the valid RulePackage or RuleFlow name.
BPM_BRM_016	400	Bad Request	Invalid CabinetName in URL	Enter the valid cabinet name.
BPM_BRM_017	400	Bad Request	Invalid isRuleFlow flag in URL	Enter the valid isRuleFlow flag.
BPM_BRM_018	400	Bad Request	Invalid encrypted flag in URL	Enter the valid encrypted flag.
BPM_BRM_019	400	Bad Request	Rule does not exist on the specified cabinet	Enter the valid RulePackage or RuleFlow ID.
BPM_BRM_020	400	Bad Request	Access Denied, token is incorrect	Enter the correct token for encrypted RulePackage or RuleFlow.
BPM_BRM_021	400	Bad Request	RulePackage/ Ruleflow is Disabled	Enable the RuleFlow or RulePackage.
BPM_BRM_022	500	Internal Server Error	Class not found in object model	Enter the valid entity class names.

Error Code	HTTP Status	HTTP Status Description	Error Code Description	Comments
BPM_BRM_023	400	Bad Request	Invalid Input Parameters	Enter the valid input parameters.
BPM_BRM_024	503	Service Unavailable	Service Unavailable	Service is either unavailable or slow.
BPM_BRM_025	500	Internal Server Error	Failed to delete key	Contact your Newgen representative.
BPM_BRM_026	409	Conflict	Server uses different cabinet	Change the cabinet configuration.
BPM_BRM_027	503	Service Unavailable	Redis server is unavailable	Contact your Newgen representative.
BPM_BRM_028	503	Service Unavailable	Service is unavailable	Service is either unavailable or slow.
BPM_BRM_029	400	Bad Request	Request body JSON invalid	Enter the valid entity member names.
BPM_BRM_030	400	Bad Request	Entity member does not match type	Enter the valid entity member value that matches the type.
BPM_BRM_031	400	Bad Request	Entity type not valid	Enter the valid entity member type.
BPM_BRM_032	409	Conflict	Data source is not valid	Check your database entries to ensure correct cabinet details.
BPM_BRM_033	400	Bad Request	Invalid input parameter	Enter the input value of defined type.
BPM_BRM_035	401	Unauthorized	User doesn't have rights to execute rule	Provide the rulePackage and ruleFlow rights to execute the rule.

Error Code	HTTP Status	HTTP Status Description	Error Code Description	Comments
BPM_BRM_036	404	Not Found	Rule deployment as microservice is not available, kindly redeploy the rule definition as microservice.	Rule deployment as microservice is not available, redeploy the rule definition as microservice.

Sample request and response body

Sample Request

```
[{
  "input": [
    {
      "className": "employeeDetails",
      "classIsArray": "N",
      "member": [
        {
          "memberIsArray": "N",
          "name": "employeename",
          "type": 10,
          "value": "joe"
        }
      ]
    }
  ],
  "output": [
    {
      "className": "employeeDetails",
      "classIsArray": "N",
      "member": [
        {
          "memberIsArray": "N",
          "name": "sacint",
          "type": 3,
          "value": "25"
        },
        {
          "memberIsArray": "Y",
          "name": "employeeRegions",
          "type": 10,
          "value": "East,South"
        }
      ]
```

```
    }
  ]
}
```

Sample Response

```
 {
  "rulesExecuted": [
    2,
    1,
    "1_Rule",
    "sachin_1_Test_RF_0",
    "SubRuledecRule_0",
    "engRule_If",
    "SubRuledecRule_1",
    "SubRuledecRule_2",
    "emp_1_Test_11549_Rule0",
    "emp_1_Test_RF_Rule0"
  ],
  "Status": "Success",
  "Output": {
    "employeeDetails": {
      "employeeId": 1,
      "employeeSubordinates": null,
      "employeeName": "s",
      "employeeRegions": ["East", "South", "West"]
        "s",
        "s",
        "s"
      ]
    }
  }
}
```

Sample code

The following sample code covers following scenarios:

This sample code is applicable for the getList API.

- Connecting a cabinet using OmniDocs (OD) Microservices
- Executing a query using the Omniapp Service
- Retrieving a rule list and input using API's from rule designer
- Executing a rule with rule endpoint fetching service
- Disconnecting the session using OD Service

```
package com.example.documentsSampleApplication;

import java.util.Arrays;
import java.util.HashMap;
import java.util.Iterator;
import java.util.Map;
import java.util.Map.Entry;

import org.json.JSONException;
import org.json.JSONObject;
import org.springframework.http.HttpEntity;
import org.springframework.http.HttpHeaders;
import org.springframework.http.HttpMethod;
import org.springframework.http.HttpStatus;
import org.springframework.http.MediaType;
import org.springframework.http.ResponseEntity;
import org.springframework.web.client.RestTemplate;

public class SampleApplication {

    public static final int HTTP_STATUS_CODE = 0;
    public static final int RESPONSE_BODY = 1;
    public static final int MAIN_CODE = 2;
    public static final int SUB_ERROR_CODE = 3;

    private static RestTemplate restClient;
    private static RestCallBrokerUtil restCallBrokerUtil;

    private void executeMicroserviceAPI() throws JSONException {
        System.out.println("*****************Connecting To a Cabinet*****************");
        String[] connectCabinetResponse = connectCabinet();
```

```
        JSONObject connect = new JSONObject(connectCabinetResponse[1]);
        String sessionId = connect.optString("userDBId");

        System.out.println("***************Fetching-Service******************");
        String[] fetchingService = fetchingService(sessionId);

        System.out.println("*****************Disconnecting User
Session*****************");
        String[] disconnectCabinet = disconnectCabinet(sessionId);
    }

    private String[] disconnectCabinet(String sessionId) throws JSONException {
        Map<String, String> headers = new HashMap<>();
        Map<String, String> queryParams = new HashMap<>();
        Map<String, String> pathParams = new HashMap<>();

        String httpMethod = "POST";
        String protocol = "https";
        String ip = "odmicroservices3.newgendocker.com/authentication";
        String context = "";
        String endPoint = "/omnidocs/v1/disconnect";

        String url = buildURI(protocol, ip, context, endPoint);

        JSONObject disconnectCabinetBody = new JSONObject();
        disconnectCabinetBody.put("userDBId", String.valueOf(sessionId));

        queryParams.put("cabinetName", "brmsmicroservice");

        String[] response = makeCall(url, httpMethod, headers, queryParams, pathParams,
                disconnectCabinetBody.toString());

        System.out.println(response[1]);
        return response;
    }

    private String[] fetchingService(String sessionId) throws JSONException {
        Map<String, String> headers = new HashMap<>();
        Map<String, String> queryParams = new HashMap<>();
        Map<String, String> pathParams = new HashMap<>();

        String httpMethod = "GET";
        String protocol = "https";
        String ip = "brmsmicroservices.newgendocker.com";
        String context = "/rule-endpoints-fetching-service-context";
        String endPoint = "/brms/getList";

        String url = buildURI(protocol, ip, context, endPoint);

        JSONObject uploadFetchingServiceBody = new JSONObject();

        headers.put("Authorization", sessionId);
        headers.put("Tenant-Id", "brmsmicroservice");
```

```
        String[] response = makeCall(url, httpMethod, headers, queryParams, pathParams,
                uploadFetchingServiceBody.toString());

        System.out.println(response[1]);
        return response;
    }

    private String[] connectCabinet() throws JSONException {
        Map<String, String> headers = new HashMap<>();
        Map<String, String> queryParams = new HashMap<>();
        Map<String, String> pathParams = new HashMap<>();

        String httpMethod = "POST";
        String protocol = "https";
        String ip = "odmicroservices3.newgendocker.com/authentication";
        String context = "";
        String endPoint = "/omnidocs/v1/connect";

        String url = buildURI(protocol, ip, context, endPoint);

        JSONObject connectInputBody = new JSONObject("{\"userPassword\":\"system123#\",
\"userName\":\"ritika\",\"locale\":\"en-US\",\"userExist\":\"N\"}");

        queryParams.put("cabinetName", "brmsmicroservice");

        String[] response = makeCall(url, httpMethod, headers, queryParams, pathParams,
                connectInputBody.toString());

        System.out.println(response[1]);
        return response;
    }

    public static void main(String[] args) throws JSONException {
        Long start = System.currentTimeMillis();
        SampleApplication restClient = new SampleApplication();
        restClient.executeMicroserviceAPI();
        Long end = System.currentTimeMillis();
        System.out.println("Overall time taken: " + (end - start));
    }

    public String buildURI(String protocol, String ip, String context, String endPoint)
{
        StringBuilder uri = new StringBuilder();
        uri.append(protocol).append("://").append(ip).append(context).append(endPoint);
        return uri.toString();
    }

    public static String[] makeCall(String uri, String method, Map<String, String>
headerParams,
            Map<String, String> queryParams, Map<String, String> pathParams, String
requestBody)
            throws JSONException {
        Long start = System.currentTimeMillis();
```

```
        HttpHeaders headers = new HttpHeaders();
        restClient = WFRestTemplateConfig.getRestTemplate();
        restCallBrokerUtil = new RestCallBrokerUtil();
        ResponseEntity<String> responseEntity = new
ResponseEntity<String>(HttpStatus.OK);
        String[] response = new String[4]; // 1 - HttpStatus, 2 - Response body, 3 -
Main code, 4 - Sub-error code
        JSONObject responseBody;

        try {
            String uri = restCallBrokerUtil.buildURI(protocol, ip, port, endPoint);
            HttpMethod httpMethod = HttpMethod.resolve(method);
            restCallBrokerUtil.populateHttpHeaders(headerParams, headers);

            if (queryParams != null && queryParams.size() > 0) {
                uri = restCallBrokerUtil.populateQueryParams(uri, queryParams);
            }

            HttpEntity<String> httpEntity = new HttpEntity<>(requestBody, headers);
            responseEntity = restClient.exchange(uri, httpMethod, httpEntity,
String.class, pathParams);
        } catch (Exception exception) {
            System.out.println("Exception block started -- ");
            exception.printStackTrace();

            String trimmedResponseBody;
            int posOfSeparator = exception.getMessage().indexOf(':');
            trimmedResponseBody = exception.getMessage().substring(posOfSeparator +
1).trim();
            System.out.println("trimmedResponseBody -- " + trimmedResponseBody);

            if (!("").equals(trimmedResponseBody.trim())) {
                System.out.println("Checking response body -- ");
                trimmedResponseBody = trimmedResponseBody.substring(1).substring(0,
                        trimmedResponseBody.length() - 2);
                response[HTTP_STATUS_CODE] = exception.getMessage().substring(0,
posOfSeparator).trim();
            }

            response[RESPONSE_BODY] = trimmedResponseBody;
            responseBody = new JSONObject(trimmedResponseBody);
            response[MAIN_CODE] = responseBody.optString("MainCode");
            response[SUB_ERROR_CODE] = responseBody.optString("SubErrorCode");
            System.out.println(Arrays.toString(response));
            exception.printStackTrace();
            return response;
        }

        Long end = System.currentTimeMillis();
        System.out.println("API execution time in milliseconds: " + (end - start));
        response[HTTP_STATUS_CODE] = responseEntity.getStatusCode().toString();
        response[RESPONSE_BODY] = responseEntity.getBody();
        responseBody = new JSONObject(responseEntity.getBody().toString());
```

```
        response[MAIN_CODE] = responseBody.optString("MainCode");
        response[SUB_ERROR_CODE] = responseBody.optString("SubErrorCode");
        return response;
    }
}

class RestCallBrokerUtil {

    public RestCallBrokerUtil() {
    }

    public void populateHttpHeaders(Map<String, String> headerParameters, HttpHeaders
headers) {
        Iterator<Entry<String, String>> iter = headerParameters.entrySet().iterator();
        while (iter.hasNext()) {
            Map.Entry<String, String> header = iter.next();
            headers.add(header.getKey(), header.getValue());
        }
        if (headers != null) {
            headers.setContentType(
                    headers.getContentType() == null ? MediaType.APPLICATION_JSON :
headers.getContentType());
        } else {
            // Exception
        }
    }

    public String populateQueryParams(String uri, Map<String, String> queryParams) {
        StringBuilder uriWithQueryParams = new StringBuilder(uri);
        uriWithQueryParams.append("?");
        Iterator<Entry<String, String>> iter = queryParams.entrySet().iterator();
        while (iter.hasNext()) {
            Map.Entry<String, String> queryParam = iter.next();
            uriWithQueryParams

  .append(queryParam.getKey()).append("=").append(queryParam.getValue()).append("&");
        }
        uriWithQueryParams.deleteCharAt(uriWithQueryParams.length() - 1);
        uri = uriWithQueryParams.toString();
        return uri;
    }

    public String buildURI(String protocol, String ip, String port, String endPoint) {
        StringBuilder uri = new StringBuilder();

uri.append(protocol).append("://").append(ip).append("/").append(port).append(endPoint);
        return uri.toString();
    }
}

class WFRestTemplateConfig {
```

```
    private static RestTemplate restTemplate;

    public WFRestTemplateConfig() {
    }

    public static RestTemplate getRestTemplate() {
        if (restTemplate == null) {
            restTemplate = new RestTemplate();
        }
        return restTemplate;
    }

    public void setRestTemplate(RestTemplate restTemplate) {
        WFRestTemplateConfig.restTemplate = restTemplate;
    }
}
```