

## Exercise 1.1 - Manage your microservice built using SAP Cloud Application Programming

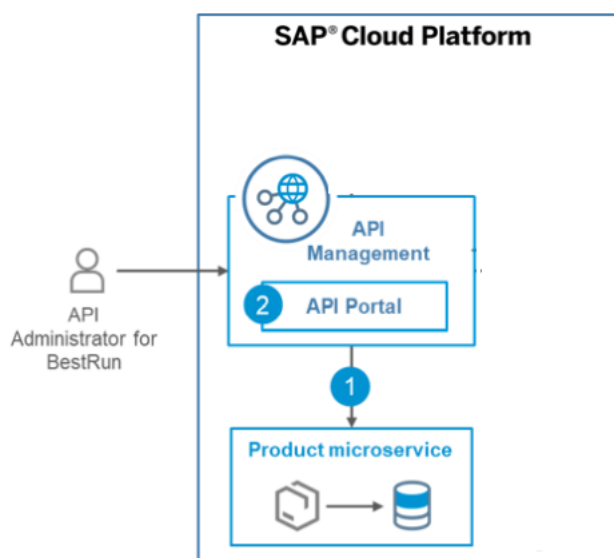
### Overview

BestRun AG plans to build modern applications like chatbots, mobile applications and web applications for their employees, business users and plans to use APIs to collaborate and co-innovate with their partners. They are worried that as the usage of their chatbots and mobile applications would grow, the load on their business applications would grow. Scaling these business applications to meet the high-volume load from the chatty applications would be costly for them in the long run and, they fear that it would increase the API response time. BestRun AG plans to replicate some of their data from business applications to SAP HANA services on cloud and then expose these data as microservices running on cloud

IT team of BestRun AG uses SAP Cloud Application Programming Model to quickly create business applications based on their product domain model. They also plan to use SAP Cloud Platform API Management to manage this microservice and apply the BestRun IT API governance policy.

In this exercise you will configure the following for BestRun AG :-

1. *Manage BestRunAG microservice using **SAP Cloud Platform API Management**.*
2. *Create an API Proxy from a microservice already built by the BestRun IT team. (Please note this microservice is already created in advance, the scope of the creation is beyond the exercise.)*
3. *Apply the BestRun IT governance policy of Quota, Spike Arrest and Verify API Key to the product service microservice*



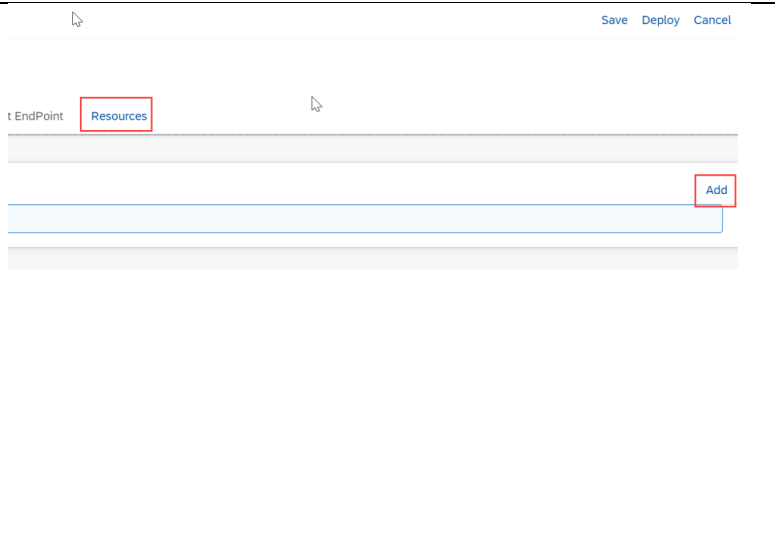
Step	Explanation	Screenshot
1	<p>Create API Proxy for your product Microservices</p> <p>To create APIProxy, launch APIPortal and Click on <b>Create</b></p>	<p>The screenshot shows the SAP Integration Suite API Portal interface. The top navigation bar includes 'SAP Integration Suite' and 'API Portal'. The main section is titled 'Develop' and contains the text 'Create APIs, Products, import Policy Templates and view Applications here.' Below this, there are tabs for 'APIs (2)', 'Products (0)', 'Applications (0)', and 'Policy Templates (2)'. A 'Create' button is highlighted with a red box, and an 'Import AP' button is visible next to it. The interface also shows a table with columns for 'Name', 'Title', and 'Status'.</p>

2	<p>You can create an API Proxy by directly providing your API URL.</p> <p>Click <input type="radio"/> URL to provide your microservice URL details.</p>	<p>Create API</p> <p>Select: <input checked="" type="radio"/> API Provider <input type="radio"/> API Proxy <input type="radio"/> URL</p> <p>API Provider: * <input type="text" value="Select"/> <input checked="" type="checkbox"/> Link API Provider</p> <p>URL * <input type="text"/></p> <p>API Details</p> <p>Name: * <input type="text"/></p> <p>Title: * <input type="text"/></p> <p>API State: * <input type="text" value="Active"/></p> <p>Host Alias: * <input type="text" value="e3c561datrial-trial.integrationsuitetrial-apim.eu10.hana.ondemand.com"/></p>
3	<p>In this exercise, you would be using the microservice already built by the BestRun IT team using SAP Cloud Application Programming model.</p> <p>Enter following values in fields &amp; click on <b>Create</b></p> <p>URL : <a href="https://dmzca0qiv9fn5be4-productsrv-srv.cfapps.eu10.hana.ondemand.com/catalog/">https://dmzca0qiv9fn5be4-productsrv-srv.cfapps.eu10.hana.ondemand.com/catalog/</a></p> <p>Name : <b>BestRunProductsAPIs</b></p> <p>Title : <b>APIs to get products and product availability</b></p> <p>API Base Path : <b>/bestrun/productservice</b></p>	<p>Create API</p> <p>Select: <input type="radio"/> API Provider <input type="radio"/> API Proxy <input checked="" type="radio"/> URL</p> <p>URL * <input type="text" value="https://dmzca0qiv9fn5be4-productsrv-srv.cfapps.eu10.hana.ondemand.com/catalog/"/></p> <p>API Details</p> <p>Name: <input type="text" value="BestRunProductsAPIs"/></p> <p>Title: * <input type="text" value="APIs to get products and product availability"/></p> <p>API State: * <input type="text" value="Active"/></p> <p>Host Alias: * <input type="text" value="e3c561datrial-trial.integrationsuitetrial-apim.eu10.hana.ondemand.com"/></p> <p>API Base Path: * <input type="text" value="bestrun/productservice"/></p> <p>Version: <input type="text"/></p> <p>Service Type: <input type="text" value="REST"/></p> <p><b>Create</b></p>
4	<p>In the <b>Overview</b> tab enter " Get information about products, price and product availability "</p> <p>as your API Proxy Description</p>	<p>SAP Integration Suite API Portal</p> <p>Create API</p> <p> BestRunProductsAPIs</p> <p><b>Overview</b> Proxy EndPoint Target EndPoint Resources</p> <p>Active</p> <p>Description: <input type="text"/></p> <p><b>B I U S</b>  Verdana 11pt <b>A</b></p> <p> </p> <p>Get information about products, price and product availability</p>

5

Based on your added resources, the path, operations field of the API Proxy definition in OpenAPI specification format will be generated. Furthermore, the corresponding conditional flows based on your added resources would be added to the policies flow. You would be able to view the added conditional flows from Policy Designers

Click on **Resources** to navigate to Resources tab and click on **Add** to manually add resource

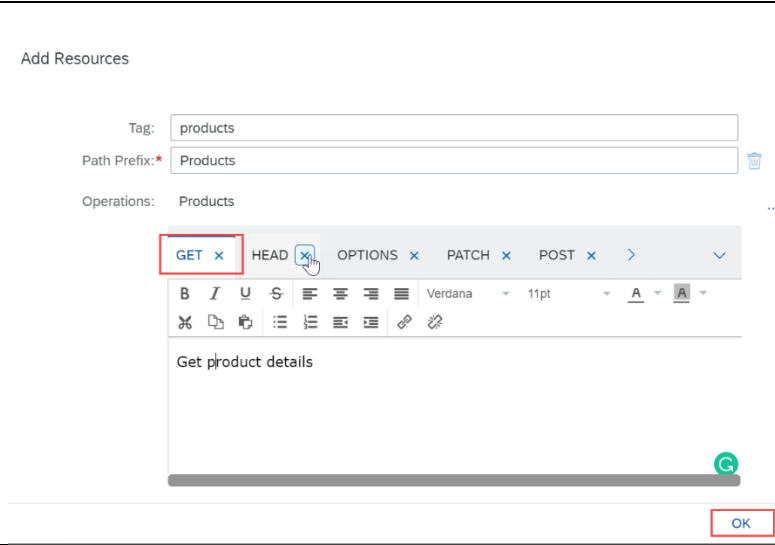


6

Enter **Products** as the Path Prefix, you can optionally set the tags to products To enable API access to only GET call, delete all operations other than GET by clicking on **X** next to Operation name

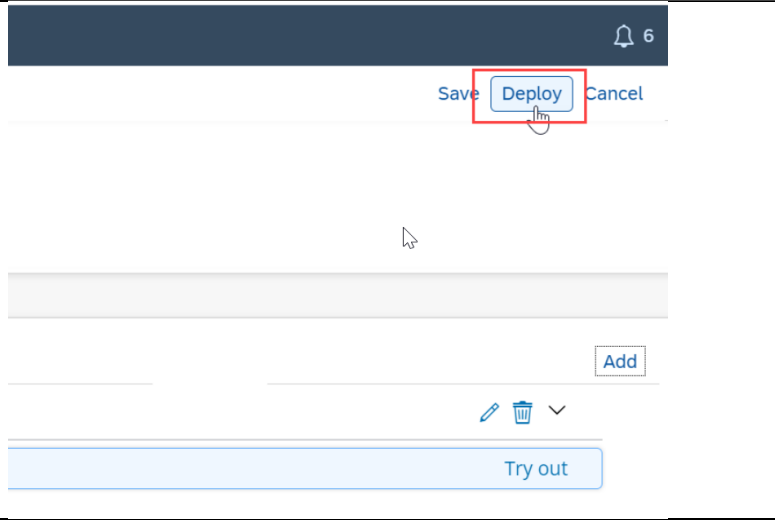
Enter **Get Product Details** in Description of Get operation


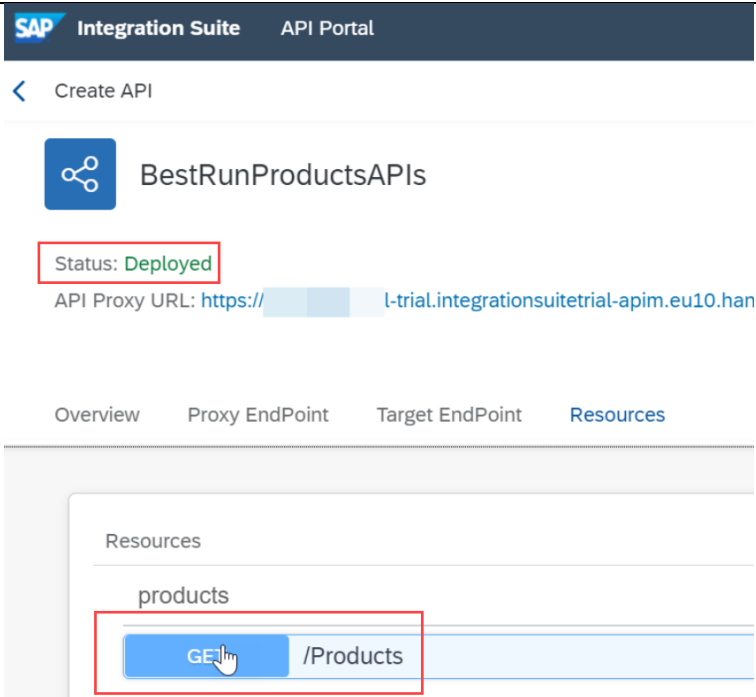
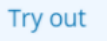
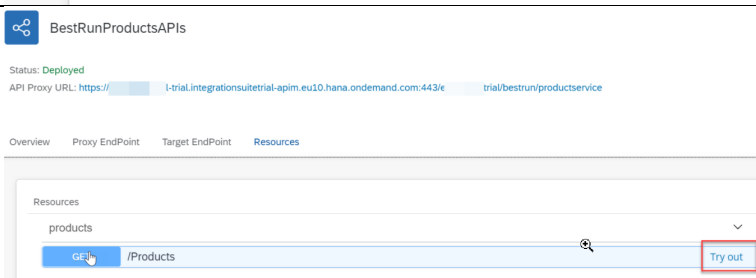

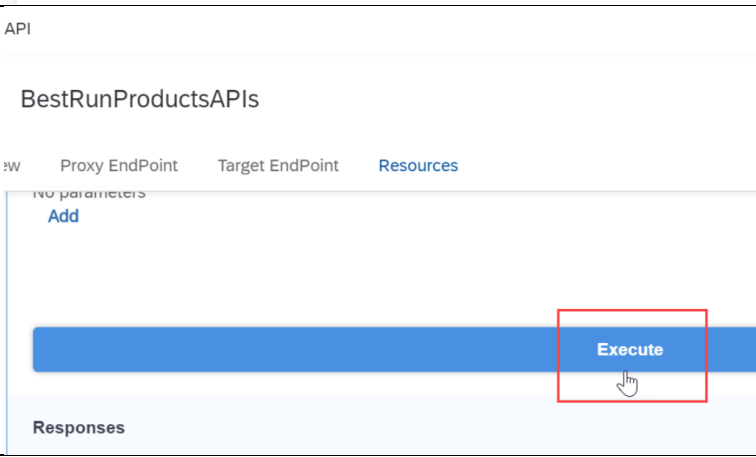
Now, click on the **OK** to add resources to API Proxy



7

Click on **Deploy** to activate the API Proxy to your microservice.



8	<p>APIProxy is deployed, check <b>Status: Deployed</b></p> <p>Click  /Products to view the API documentation and test the select resources.</p>	
9	<p>Click  to test the selected resource</p>	
10	<p>Click  to test the API Proxy</p>	

11

The response from your microservice can viewed inline in the Server response section

SAP

Integration Suite

API Portal

<

Create API

BestRunProductsAPIs

Overview

Proxy EndPoint

Target EndPoint

Resources

Execute

Responses

Server response

Code

Details

200

Response body

```
{
  "@odata.context": "$metadata#Products",
  "@odata.metadataEtag": "W/\"bVaa5C59nKbbeEh/dKc1TzsyNE3Zx6vitzgm1ZhCAGM=\\",
  "value": [
    {
      "productId": "HT-1010",
      "category": "Notebooks",
      "productName": "Notebook Professional 15",
      "description": "Notebook Professional 15 with 2,80 GHz quad core, 15 inch Mu
      RM/-RAM),Windows 8 Pro",
      "supplierID": "0100000005",
      "supplierName": "TECUM",
      "weightMeasure": 4.3,
    }
  ]
}
```

Congratulations, you have successfully created API Proxy to manage your microservice. In the next exercise you will be adding the BestRun IT governance policies to your microservice

Continue to **Exercise 1.2 - Add Corporate Governance Policy**