

[Explore](#)

AI Studio - Published Service 1.0.0 OAS 3.0

Servers

<https://localhost:62673>

invocations MLflow model invocation



POST **/invocations** Do a Model Invocation



description

Parameters

Cancel

Reset

No parameters

Request body required

application/json

Input from model signature

```
{
  "inputs": {
    "question": [
      "Write Python code to scrape a website and extract all links"
    ]
  },
  "params": {}
}
```

Execute**Clear**

Responses

Curl

```
curl -X 'POST' \
  'https://localhost:62673/invocations' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
    "inputs": {
      "question": [
        "Write Python code to scrape a website and extract all links"
      ]
    },
    "params": {}
  }'
```

Request URL

https://localhost:62673/invocations

Server response

Code Details

200

Response body

```
{
  "predictions": [
    {
      "result": "import requests\nfrom bs4 import BeautifulSoup\ndef scrape_website(url):\n    headers = {'User-Agent':\n'Mozilla/5.0'}\n    response = requests.get(url, headers=headers)\n    soup = BeautifulSoup(response.text, 'html.parse\nr')\n    links = []\n    for link in soup.find_all('a'):\n        try:\n            href = link.get('href')\n            if href and not href.startswith('#'):\n                links.append(href)\n        except Exception as e:\n            print(f\"An error occurred: {str(e)}\")\n    return links\nurl = 'https://www.example.com'\nprint(scrape_website(url))\n# Output will be a list of all extracted URLs. ''' import requests\nfrom bs4 import BeautifulSoup\ndef scrape_website\n(url):\n    headers = {'User-Agent': 'Mozilla/5.0'}\n    response = requests.get(url, headers=headers)\n    soup = Beau\ntifulSoup(response.text, 'html.parser')\n    links = []\n    for link in soup.find_all('a'):\n        try:\n            href = link.get('href')\n            if href and not href.startswith('#'):\n                links.append(href)\n        except Exception as e:\n            print(f\"An error occurred: {str(e)}\")\n    return links\nurl = 'https://www.exam\nple.com'\nprint(scrape_website(url))\n# Output will be a list of all extracted URLs. ''' import requests\nfrom bs4 imp\nort BeautifulSoup\ndef scrape_website(url):\n    headers = {'User-Agent': 'Mozilla/5.0'}\n    response = requests.get(u\nrl, headers=headers)\n    soup = BeautifulSoup(response.text, 'html.parser')\n    links = []\n    for link in soup.find\n_all('a'):\n        try:\n            href = link.get('href')\n            if href and not href.startswith('#'):\n                links.append(href)\n        except Exception as e:\n            print(f\"An error occurred: {str(e)}\")\n    return li\nks\nurl = 'https://www.example.com'\nprint(scrape_website(url))\n# Output will be a list of all extracted URLs. ''' i\nimport requests\nfrom bs4 import BeautifulSoup\ndef scrape_website(url):\n    headers = {'User-Agent': 'Mozilla/5.0'}\n    response = requests.get(url, headers=headers)\n    soup = BeautifulSoup(response.text, 'html.parser')\n    links = []\n    for link in soup.find_all('a'):\n        try:\n            href = link.get('href')\n            if href and not href.st\nartswith('#'):\n                links.append(href)\n        except Exception as e:\n            print(f\"An error occur\nred: {str(e)}\")\n    return links\nurl = 'https://www.example.com'\nprint(scrape_website(url))\n# Output will be a li\nst of all extracted URLs. ''' \nimport requests\nfrom bs4 import BeautifulSoup\ndef scrape_website(url):\n    headers\n= {'User-Agent': 'Mozilla/5.0'}\n    response = requests.get(url, headers=headers)\n    soup = Beautifu~ (r\next, 'html.parser')\n    links = []\n    for link in soup.find_all('a'):\n        try:\n            hre   |ir Download\n            href = link.get('href')\n            if href and not href.startswith('#'):\n                links.append(href)\n            except Exception a
```

Response headers

```
access-control-allow-origin: https://localhost:62673
content-length: 4551
content-type: application/json
date: Fri,01 Aug 2025 04:19:30 GMT
server: envoy
vary: Origin
x-envoy-upstream-service-time: 46530
```

Responses

Code	Description	Links
200	<p>Successful operation</p> <p>Media type</p> <div>application/json</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "result": "string"}</pre>	No links

Schemas

ModelInput

inputs

{

question

repository_url

[...]

[string]

}

params

{...}

ModelOutput

ModelOutput