

CKAN Data APIs

ນຳໃຫ້ CKAN Data APIs
Create
Update
Read
Delete



Tools ที่ใช้ในการทดสอบ CKAN Data APIs



POSTMAN



Visual Studio Code

ทดสอบ CKAN Data APIs

Method GET คือการเรียกดูข้อมูล (Read)

Ckan.logic.action.get

Get คือการดึงข้อมูลจากฐานข้อมูล และ Method Get จะส่งข้อมูล
แบบต่อท้าย URL

1. แสดงรายละเอียดของชุดข้อมูล (Package)

Get	http://{domain}/api/3/action/package_show	1
Authorization	Require	2
Request	{"id":"{package_id}"}	3



วิธีการทดสอบ API method Get ด้วย Postman

The screenshot shows the Postman interface with the following configuration:

- Method:** GET
- URL:** http://domain/api/3/action/package_show?package_id
- Params:** A table with one row: **Id** (Key) and **package_id** (Value).
- Headers:** A table with one row: **Authorization** (Key) and **API Key** (Value).
- Send** button highlighted with a red box and number 1.
- Query Params** section is visible below the main fields.

2. แสดงรายละเอียดชุดข้อมูลตามที่ค้นหา

Get	http://{domain}/api/3/action/package_search
Authorization	Require
Request	{"include_private":"true", "q":"owner_org:{organization_id}"}

3. แสดงรายละเอียดข้อมูลและทรัพยากร (Resource)

Get	http://{domain}/api/3/action/resource_show
Authorization	Require
Request	{"id":"{resource_id}"}

4. แสดงรายละเอียดของหน่วยงาน

Get	http://{domain}/api/3/action/organization_show
Authorization	Require
Request	{"id":"{organization_id}"}

5. แสดง Organization list

Get	<code>http://{domain}/api/3/action/organization_list</code>
Authorization	Require
Request	<code>{"order_by":"name", "limit":"10"}</code>

6. แสดงรายละเอียดหน่วยงานของ User

Get	http://{domain}/api/3/action/organization_list_for_user
Authorization	Require
Request	{"id":"{user_id}", "permission":"create_dataset", "include_dataset_count":"true"}

7. แสดงรายละเอียดของ User

Get	http://{domain}/api/3/action/user_show
Authorization	Require
Request	{"id": "{user_id}", "include_datasets": "true", "all_fields": "true"}

8. แสดง User list

Get	<code>http://{domain}/api/3/action/user_list</code>
Authorization	Require
Request	<code>{"q":"{user_name}", "email":"{email}","order_by":"name", "all_fields":true}</code>

9. แสดงรายละเอียดของ Activity

Get	http://{domain}/api/3/action/activity_show
Authorization	Require
Request	{"id": "{activity_id}", "include_data": "true"}

10. แสดงรายละเอียดข้อมูลตามประเภทที่ค้นหาด้วย activity id

Get	http://{domain}/api/3/action/activity_data_show
Authorization	Require
Request	{"id": "{activity_id}", "object_type": "package"}

ทดสอบ CKAN Data APIs

Method Post คือการเพิ่มข้อมูล (Create)

Ckan.logic.action.create

Create คือการสร้างข้อมูล โดยการส่งข้อมูลไปสร้างข้อมูลลงในฐานข้อมูล และต้องส่งข้อมูลที่มีการ Validation ไปด้วยทุกครั้ง

1. เพิ่มหน่วยงาน

Post	http://{domain}/api/3/action/organization_create	1
Authorization	Require	2
Request	{"name": "{organization_name}", "title": "ตัวอย่างเพิ่มหน่วยงาน", "description": "ทดสอบการสร้างหน่วยงาน", "state": "active"}	3
Response	"help": "http://{domain}/api/3/action/help_show?name=organization_create", "success": true, "result": {ข้อมูลหน่วยงานที่สร้างเรียบร้อยแล้ว}}	4



วิธีการทดสอบ API method Post ด้วย Postman

1

POST http://domain /api/3/action/organization_create Send

Params Authorization Headers (10) Body Pre-request Script Tests Settings

Headers 9 hidden

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> Authorization	API Key			

Params Authorization Headers (10) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

1 {"name": "testorg2",
2 "title": "ตัวอย่างเพิ่มนิวยงาน",
3 "description": "ทดสอบการสร้างนิวยงาน",
4 "state": "active"
5 }

3

Body Cookies (1) Headers (6) Test Results Status: 200 OK

Pretty Raw Preview Visualize JSON

1 {
2 "help": "http://domain /api/3/action/help_show?name=organization_create",
3 "success": true,
4 "result": [
5 "display_name": "ตัวอย่างเพิ่มนิวยงาน",
6 "description": "ทดสอบการสร้างนิวยงาน",

4

2. เพิ่ม User

Post	<code>http://{domain}/api/3/action/user_create</code>
Authorization	Require
Request	<code>{"name": "{user_name}", "email": "{email}", "password": "{password}", "fullname": "{name surname}"}</code>
Response	<code>"help":"http://{domain}/api/3/action/help_show?name=user_create", "success": true, "result": {ข้อมูล User ที่สร้างเรียบร้อยแล้ว}</code>

3. เพิ่มสมาชิก (User) ในหน่วยงาน

Post	<code>http://{domain}/api/3/action/organization_member_create</code>
Authorization	Require
Request	<pre>{"id" : "{organization_id}", "username": "{user_name}", "role": "member" {member, editor, or admin} }</pre>
Response	<pre>"help":"http://{domain}/api/3/action/help_show?name=organization_member_create", "success": true, "result": {ข้อมูลสมาชิกที่เพิ่มเรียบร้อยแล้ว}</pre>

4. เพิ่มชุดข้อมูล (Package)

Post	http://{domain}/api/3/action/package_create
Authorization	Require
Request	{"name": "{package_name}", "data_type": "ข้อมูลระเบียน", "data_source": "{แหล่งที่มา}", "owner_org": "{organization_name}", "data_format": ["อินๆ"], "data_format_other": "Excel", "data_category": "ข้อมูลสาธารณะ", "objective": ["ไม่ทราบ"], "title": "เพิ่มชุดข้อมูล", "maintainer": "{name}", "maintainer_email": "{email}", "tag_string": ["คำสำคัญ"], "notes": "ทดสอบการเพิ่มชุดข้อมูลด้วย API", "update_frequency_unit": "ตามเวลาจริง", "geo_coverage": "ไม่มี", "license_id": "License not specified", "private": "true"}
Response	"help": " http://{domain}/api/3/action/help_show?name=package_create ", "success": true, "result": {ชุดข้อมูลที่สร้างเรียบร้อยแล้ว}}

5. เพิ่มข้อมูลและทรัพยากร (Resource)

Post	<code>http://{domain}/api/3/action/resource_create</code>
Authorization	Require
Request	<pre>{"package_id": "{package_id}", "name": "ตัวอย่างเพิ่ม resource", "description": "ทดสอบการสร้าง resource", "format": "XLSX", "state": "active"}</pre>
Response	<pre>"help": "http://{domain}/api/3/action/help_show?name=resource_create", "success": true, "result": {ข้อมูลและทรัพยากรที่สร้างเรียบร้อยแล้ว}}</pre>

ทดสอบ CKAN Data APIs

แก้ไขข้อมูล (Update)

(ใน CKAN จะใช้ Method Post)

Ckan.logic.action.update

Update คือการแก้ไขข้อมูลทั้งหมด จำเป็นต้องส่งข้อมูลเดิมกลับไปทั้งหมดเพื่อ Update ข้อมูลใหม่ทั้งหมด

1. แก้ไขรายละเอียดในชุดข้อมูล (Package)

Post	http://{domain}/api/3/action/package_update
Authorization	Require
Request	{"id": "{package_id}", "name": "{package_name}", "data_type": "ข้อมูลระเบียน", "data_source": "{แหล่งที่มา}", "owner_org": "{organization_name}", "data_format": ["อินๆ"], "data_format_other": "Excel", "data_category": "ข้อมูลสาธารณะ", "objective": ["ไม่ทราบ"], "title": "แก้ไขชุดข้อมูล", "maintainer": "{name}", "maintainer_email": "{email}", "tag_string": ["คำสำคัญ"], "notes": "ทดสอบการแก้ไขชุดข้อมูลด้วย API", "update_frequency_unit": "ตามเวลาจริง", "geo_coverage": "ไม่มี", "license_id": "License not specified", "private": "true"}
Response	"help": " http://{domain}/api/3/action/help_show?name=package_update ", "success": true, "result": {ชุดข้อมูลที่แก้ไขเรียบร้อยแล้ว}}

2. แก้ไขรายละเอียดของข้อมูลและทรัพยากร (Resource)

Post	<code>http://{domain}/api/3/action/resource_update</code>
Authorization	Require
Request	<pre>{"id" : "{resource_id}", "package_id" : "{package_id}", "name" : "ตัวอย่างการแก้ไข resource", "description": "ทดสอบการแก้ไข resource", "format" : "XLSX", "state" : "active"}</pre>
Response	<pre>"help":"http://{domain}/api/3/action/help_show?name=resource_update", "success": true, "result": {ข้อมูลและทรัพยากรที่แก้ไขเรียบร้อยแล้ว}}</pre>

3. แก้ไขรายละเอียดข้อมูล User

Post	<code>http://{domain}/api/3/action/user_update</code>
Authorization	Require
Request	<code>{"id": "{user_id}", "email": "{email}", "fullname": "{name surname}"} "help": "http://{domain}/api/3/action/help_show?name=user_update", "success": true, "result": {ข้อมูล User ที่แก้ไขเรียบร้อยแล้ว}</code>
Response	<code>"help": "http://{domain}/api/3/action/help_show?name=user_update", "success": true, "result": {ข้อมูล User ที่แก้ไขเรียบร้อยแล้ว}</code>

4. แก้ไขรายละเอียดข้อมูลหน่วยงาน

Post	http://{domain}/api/3/action/organization_update
Authorization	Require
Request	{"id": "{organization_id}", "name": "{organization_name}", "title": "ตัวอย่างการแก้ไขหน่วยงาน", "description": "ทดสอบการแก้ไขหน่วยงาน", "state": "active"}
Response	"help": "http://{domain}/api/3/action/help_show?name=organization_update", "success": true, "result": {ข้อมูลหน่วยงานที่แก้ไขเรียบร้อยแล้ว}}

5. Generate user API key

Post	<code>http://{domain}/api/3/action/user_generate_apikey</code>
Authorization	Require
Request	<code>{"id" : "{user_id}"}</code>
Response	<code>"help":"http://{domain}/api/3/action/help_show?name=user_generate_apikey", "success": true, "result": {ข้อมูล User ที่สร้าง Api Key เรียบร้อยแล้ว}</code>

ทดสอบ CKAN Data APIs

แก้ไขข้อมูลบางส่วน (PATCH)

(ใน CKAN จะใช้ Method Post)

Ckan.logic.action.patch

Patch คือการแก้ไขข้อมูลเฉพาะบางส่วนเท่านั้น ไม่จำเป็นต้องล้าง
ข้อมูลเดิมกลับไปทั้งหมด

1. แก้ไขรายละเอียดในชุดข้อมูลบางส่วน

Post	<code>http://{domain}/api/3/action/package_patch</code>
Authorization	Require
Request	<code>{"id" : "{package_id}", "title" : "ทดสอบ package patch"}</code>
Response	<code>"help":"http://{domain}/api/3/action/help_show?name=package_patch", "success": true, "result": {"รายละเอียดชุดข้อมูลที่แก้ไขเรียบร้อยแล้ว"}}</code>

2. แก้ไขข้อมูลและทรัพยากรบагส่วน

Post	http://{domain}/api/3/action/resource_patch
Authorization	Require
Request	{"id" : "{resource_id}", "name" : "ทดสอบ resource patch"}
Response	"help":" http://{domain}/api/3/action/help_show?name=resource_patch ", "success": true, "result": {รายละเอียดข้อมูลและทรัพยากรที่แก้ไขเรียบร้อยแล้ว}}

3. แก้ไขรายละเอียดข้อมูลในหน่วยงานบางส่วน

Post	http://{domain}/api/3/action/organization_patch
Authorization	Require
Request	{"id" : "{organization_id}", "title": "ทดสอบ organization patch"}
Response	"help":" http://{domain}/api/3/action/help_show?name=organization_patch ", "success": true, "result": {"รายละเอียดหน่วยงานที่แก้ไขเรียบร้อยแล้ว"}}

ทดสอบ CKAN Data APIs

ลบข้อมูล (Delete)

(ใน CKAN จะใช้ Method Post)

Ckan.logic.action.delete

Delete คือการลบข้อมูล โดยข้อมูลจะยังไม่ถูกลบออกจากฐานข้อมูล
แต่จะถูกปรับสถานะเป็น deleted

1. ลบข้อมูลและทรัพยากร (Resource)

Post	<code>http://{domain}/api/3/action/resource_delete</code>
Authorization	Require
Request	<code>{"id" : "{resource_id}"}</code>
Response	<code>"help":"http://{domain}/api/3/action/help_show?name=resource_delete", "success": true, "result": null</code>

2. លប់ចុះឈ្មោះមូល (Package)

Post	<code>http://{domain}/api/3/action/package_delete</code>
Authorization	Require
Request	<code>{"id" : "{package_id}"}</code>
Response	<code>"help":"http://{domain}/api/3/action/help_show?name=package_delete", "success": true, "result": null</code>

3. ลบสมาชิกออกจากหน่วยงาน

Post	http://{domain}/api/3/action/member_delete
Authorization	Require
Request	{"id" : "{organization_id}", "object" : "{user_id}", "object_type" : "user"}
Response	"help":" http://{domain}/api/3/action/help_show?name=member_delete ", "success": true, "result": null

4. ลบ User

Post	http://{domain}/api/3/action/user_delete
Authorization	Require
Request	{"id" : "{user_id}"}
Response	"help":" http://{domain}/api/3/action/help_show?name=user_delete ", "success": true, "result": null

5. ลบหน่วยงาน

Post	http://{domain}/api/3/action/organization_delete
Authorization	Require
Request	{"id" : "{organization_id}"}
Response	"help":" http://{domain}/api/3/action/help_show?name=organization_delete ", "success": true, "result": null

เอกสารอ้างอิง

<https://docs.ckan.org/en/2.9/api/#action-api-reference>