CodyNick MBDB Guide

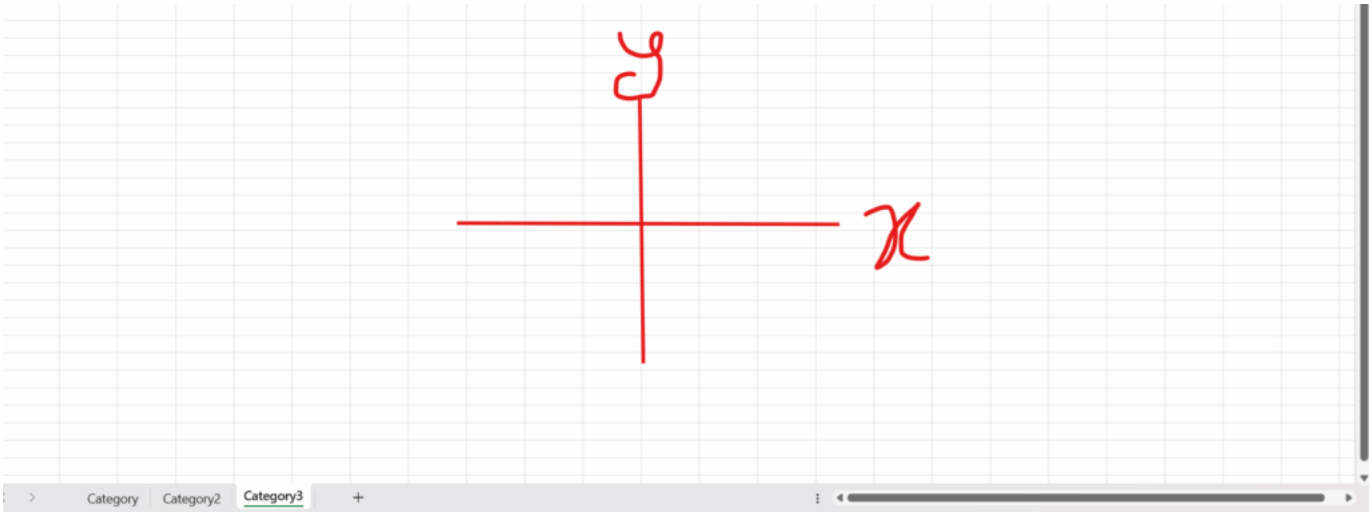
Introduction

The CodyNick MBDB enables you to store data permanently and use it across different CodyNick codes, or at different times.

To get started, go to https://mbdb.codynick.com and obtain an instance code. Make sure to keep it in a safe place other than in your code if you want to access the data later.

Each MBDB Entry has 5 parameters:

- Instance: Identifies the user or the project. Is signified by a random string.
- Category: categories are similar to tables in databases or seperate sheets in an excel file. They allow you to have multiple datasets within the same instance. This can be any string.
- x: The virtual horizontal axis of your table. Must be an integer
- y: The virtual vertical axis of your table. Must be an integer
- Value: The content in the cell at (x,y) in the category, in the instance. This is where you store the actual data.



Here is a general impression of what an instance would look like graphically. (in reality, they are stored in a list format for all instances, categories, and coordinates)

Instance Data Dumps

To conduct a manual instance data dump of an instance, go to https://mbdb.codynick.com and navigate to the “Instance Dump” tab. Enter your instance code, and you can see all entries made across all categories and coordinates in a list format.

MBDB Queries

All queries to MBDB should be sent to https://mbdb.codynick.com with the following GET parameters:

- q: This is an integer with signifies the type of query. There are 6 different types:
  - 1 – A cell read or write query
  - 2 – Instance generator
  - 3 – Instance Dump (text format – not suitable for processing with codes)
  - 4 – Category eraser
  - 5 – Column CSV output or sum of all number values
  - 6 – Highest and lowest y-values in the column

seperated by “,”. Does not account for deleted values in the middle (there may be gaps)

- Instance: this is your instance code as discussed earlier.
- Category: this is your category as discussed earlier
- x and y: your coordinates as discussed earlier
- value: your cell value

Here is a list of which inputs each query type requires:

q	instance	category	x	y	value	type
1	□	□	□	□	□	0:read 1:write
2	□	□	□	□	□	□
3	□	□	□	□	□	□
4	□	□	□	□	□	□
5	□	□	□	□	□	0:CSV 1:Sum
6	□	□	□	□	□	□

Note that for q=1 (read and write) you need a value even if reading, but it can be an arbitrary or empty value for reading. If you give an empty value for a write request, the record will be deleted and won’t be considered by q=6 request if it was an extremum.

Query Outputs

For requests q=1,4,5,6:

{

“status” : ###,

“value”: OUTPUT_OR_MESSAGE

}

For request q=2:

{

“status” : ###,

“instance”: INSTANCE_CODE

}

For requests q=3:

{

“status” : ###,

“raw”: RAW_DATA

}