

- [Issue ref JSON format](#)
  - [PysonDB required format](#)
  - [Pandas & table-reader plugin required format](#)
  - [Possible quick-dev solutions](#)
    - [1. let PysonDB format and convert before load table with table-reader](#)

## Issue ref JSON format

Ref database: `books_catalog.json`

### PysonDB required format

```
{
  "data": [
    {
      "id": 0,
      "code": "BCAT",
      "short_desc": "bk_tmpl",
      "description": "Sablonul implicit pentru o carte noua",
      "created_date": "2023-08-01",
      "created_by": "system",
      "notes": "Inregistrare obligatorie de la instalare sistem. Non editabila."
    }
  ]
}
```

The format is a *dict* with key *"data"* which contain *array with records*.

### Pandas & table-reader plugin required format

The accepted format (for `orient='records'` Pandas spec of `read_json()` function) is:

```
[
  {
    "id": 0,
    "code": "BCAT",
    "short_desc": "bk_tmpl",
    "description": "Sablonul implicit pentru o carte noua",
    "created_date": "2023-08-01",
    "created_by": "system",
    "notes": "Inregistrare obligatorie de la instalare sistem. Non editabila."
  }
]
```

## Possible quick-dev solutions

### 1. let PysonDB format and convert before load table with `table-reader`

- before execute
 

id	code	short_desc	description	
created_date	created_by	notes		
-----	-----	-----	-----	-----
-----	-----	-----	-----	-----
0	BCAT	bk_tmpl	Sablonul implicit pentru o carte noua	2023-08-01   system
Inregistrare obligatorie de la instalare sistem. Non editabila.				
- insert a python static code to extract from PysonDB format just key `"data"` and
  - return its value to `read_json()` function instead reading from file
  - or create a temporary file with this string saved and use it in `read_json()` function

---

Last update: September 9, 2023