

Strategy: DOCUMENTS_BY_VIEW

Use this strategy to get all documents from the view.

Allowed method

- GET

Parameters:

- `databaseName` (String `databaseName`): name of the database to read data from, use "server!!path/database.nsf" format; when omitted, current database is used
- `viewName` (String `viewName`): name of the view; mandatory

Let's assume our database has view called (`ByTopic`).

```
router.GET('topics') {
  strategy(DOCUMENTS_BY_VIEW) {
    viewName('ByTopic')
  }
  mapJson 'id', json:'id', type:'STRING', isformula:true, formula:'@DocumentUniqueID'
  mapJson 'date_created', json:'date_created', type:'DATETIME', isformula:true, formula:'@Created'
  mapJson 'topic', json:'topic', type:'STRING'
  mapJson 'author', json:'author', type:'STRING', isformula:true, formula:'@Name([CN]; @Author)'
}
```

With route defined above, URL `http://server.name/path-to/db.nsf/xsp/.xrest/topics` will return all documents from view (`ByTopic`).

Returned JSON will contain an array of documents from the view, such as:

```
[
  {
    "author": "Martin Jinoch",
    "id": "54502859C07299C7C12580D8006404F4",
    "date_created": "2017-03-03T11:01+00:00",
    "topic": "test"
  },
  {
    "date_created": "2017-03-03T11:01+00:00",
    "author": "Martin Jinoch",
    "id": "DFDBA4D1FED3D5D6C12580D8003DE42E",
    "topic": "100009"
  },
  {
    "date_created": "2017-03-03T11:01+00:00",
    "author": "Martin Jinoch",
    "id": "558351BABADEA987C12580D8003DE42F",
    "topic": "100008"
  },
  {
    "date_created": "2017-03-03T11:01+00:00",
    "author": "Martin Jinoch",
    "id": "FD3FC3BFE506F278C12580D8003DE430",
    "topic": "100007"
  },
  {
    "date_created": "2017-03-03T11:01+00:00",
    "author": "Martin Jinoch",
    "id": "259F5E702AE9D1A3C12580D8003DE431",
    "topic": "100006"
  },
  {
    "date_created": "2017-03-03T11:01+00:00",
    "author": "Martin Jinoch",
```

```

    "id": "7D275279C52C1F44C12580D8003DE432",
    "topic": "100005"
  },
  {
    "date_created": "2017-03-03T11:01+00:00",
    "author": "Martin Jinoch",
    "id": "1411C86852B57559C12580D8003DE433",
    "topic": "100004"
  },
  {
    "date_created": "2017-03-03T11:01+00:00",
    "author": "Martin Jinoch",
    "id": "393C379E0527C246C12580D8003DE434",
    "topic": "100003"
  },
  {
    "date_created": "2017-03-03T11:01+00:00",
    "author": "Martin Jinoch",
    "id": "66671AF824F5A3D3C12580D8003DE435",
    "topic": "100002"
  },
  {
    "date_created": "2017-03-03T11:01+00:00",
    "author": "Martin Jinoch",
    "id": "01ED9C403BE0ED4FC12580D8003DE436",
    "topic": "100001"
  },
  {
    "date_created": "2017-03-03T11:01+00:00",
    "author": "Martin Jinoch",
    "id": "F52C541EC8EF970EC12580D8003DE437",
    "topic": "100000"
  },
  {
    "date_created": "2017-03-03T11:01+00:00",
    "author": "Martin Jinoch",
    "id": "E044D66CFEA6735AC12580D8003F43C7",
    "topic": "10000"
  }
]

```

Related articles

- [Strategy: CUSTOM](#)
- [Strategy: DOCUMENT_BY_UNID](#)
- [Strategy: DOCUMENT_FROM_VIEW_BY_KEY](#)
- [Strategy: DOCUMENTS_BY_FORMULA](#)
- [Strategy: DOCUMENTS_BY_FORMULA_PAGED](#)
- [Strategy: DOCUMENTS_BY_SEARCH_FT](#)
- [Strategy: DOCUMENTS_BY_SEARCH_FT_PAGED](#)
- [Strategy: DOCUMENTS_BY_VIEW](#)
- [Strategy: DOCUMENTS_BY_VIEW_PAGED](#)
- [Strategy: DOCUMENTS_FROM_VIEW_BY_KEY](#)
- [Strategy: DOCUMENTS_FROM_VIEW_BY_KEY_PAGED](#)
- [Strategy: VIEWENTRIES](#)
- [Strategy: VIEWENTRIES_BY_CATEGORY](#)
- [Strategy: VIEWENTRIES_BY_CATEGORY_PAGED](#)
- [Strategy: VIEWENTRIES_PAGED](#)