

# itemprototype.get

## Description

integer/array `itemprototype.get(object parameters)`

The method allows to retrieve item prototypes according to the given parameters.

## Parameters

(object) Parameters defining the desired output.

The method supports the following parameters.

| Parameter                   | Type         | Description  |
|-----------------------------|--------------|--|
| discoveryids                | string/array | Return only item prototypes that belong to the given LLD rules.  |
| graphids                    | string/array | Return only item prototypes that are used in the given graph prototypes.   |
| hostids                     | string/array | Return only item prototypes that belong to the given hosts.  |
| inherited                   | boolean      | If set to <code>true</code> return only item prototypes inherited from a template.                                     |
| itemids                     | string/array | Return only item prototypes with the given IDs.  |
| monitored                   | boolean      | If set to <code>true</code> return only enabled item prototypes that belong to monitored hosts.                        |
| templated                   | boolean      | If set to <code>true</code> return only item prototypes that belong to templates.                                      |
| templateids                 | string/array | Return only item prototypes that belong to the given templates.  |
| triggerids                  | string/array | Return only item prototypes that are used in the given trigger prototypes.   |
| selectApplications          | query        | Return applications that the item prototype belongs to in the <code>applications</code> property.                      |
| selectApplicationPrototypes | query        | Return application prototypes linked to item prototype in <code>applicationPrototypes</code> property.                 |
| selectDiscoveryRule         | query        | Return the low-level discovery rule that the graph prototype belongs to in the <code>discoveryRule</code> property.    |
| selectGraphs                | query        | Return graph prototypes that the item prototype is used in in the <code>graphs</code> property.<br>Supports count.     |
| selectHosts                 | query        | Returns the host that the item prototype belongs to as an array in the <code>hosts</code> property.                    |
| selectTriggers              | query        | Return trigger prototypes that the item prototype is used in in the <code>triggers</code> property.<br>Supports count. |

| Parameter              | Type         | Description  |
|------------------------|--------------|--|
| selectPreprocessing    | query        | <p>Return item preprocessing options in preprocessing property.</p> <p>It has the following properties:<br/>           type - (string) The preprocessing option types:<br/>           1 - Custom multiplier;<br/>           2 - Right trim;<br/>           3 - Left trim;<br/>           4 - Trim;<br/>           5 - Regular expression matching;<br/>           6 - Boolean to decimal;<br/>           7 - Octal to decimal;<br/>           8 - Hexadecimal to decimal;<br/>           9 - Simple change;<br/>           10 - Change per second.</p> <p>params - (string) Additional parameters used by preprocessing option. Multiple parameters are separated by LF (\n)character.</p> |
| filter                 | object       | <p>Return only those results that exactly match the given filter.</p> <p>Accepts an array, where the keys are property names, and the values are either a single value or an array of values to match against.</p> <p>Supports additional filters:<br/>           host - technical name of the host that the item prototype belongs to.</p>  |
| limitSelects           | integer      | <p>Limits the number of records returned by subselects.</p> <p>Applies to the following subselects:<br/>           selectGraphs - results will be sorted by name;<br/>           selectTriggers - results will be sorted by description.</p>   |
| sortfield              | string/array | <p>Sort the result by the given properties.</p> <p>Possible values are: itemid, name, key_, delay, type and status.</p>  |
| countOutput            | boolean      | <p>These parameters being common for all get methods are described in detail in the <a href="#">reference commentary</a>.</p>  |
| editable               | boolean      |  |
| excludeSearch          | boolean      |  |
| limit                  | integer      |  |
| output                 | query        |  |
| preservekeys           | boolean      |  |
| search                 | object       |  |
| searchByAny            | boolean      |  |
| searchWildcardsEnabled | boolean      |  |
| sortorder              | string/array |  |
| startSearch            | boolean      |  |

## Return values

(integer/array) Returns either:

- an array of objects;
- the count of retrieved objects, if the countOutput parameter has been used.

## Examples

### Retrieving item prototypes from an LLD rule

Retrieve all item prototypes from an LLD rule.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "itemprototype.get",
  "params": {
    "output": "extend",
    "discoveryids": "27426"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "itemid": "23077",
      "type": "0",
      "snmp_community": "",
      "snmp_oid": "",
      "hostid": "10079",
      "name": "Incoming network traffic on $1",
      "key_": "net.if.in[en0]",
      "delay": "1m",
      "history": "1w",
      "trends": "365d",
      "status": "0",
      "value_type": "3",
      "trapper_hosts": "",
      "units": "bps",
      "snmpv3_securityname": "",
      "snmpv3_securitylevel": "0",
```

```
"snmpv3_authphrase": "",
"snmpv3_privphrase": "",
"formula": "",
"error": "",
"lastlogsize": "0",
"logtimefmt": "",
"templateid": "0",
"valuemapid": "0",
"params": "",
"ipmi_sensor": "",
"authtype": "0",
"username": "",
"password": "",
"publickey": "",
"privatekey": "",
"mtime": "0",
"flags": "0",
"interfaceid": "0",
"port": "",
"description": "",
"inventory_link": "0",
"lifetime": "30d",
"snmpv3_authprotocol": "0",
"snmpv3_privprotocol": "0",
"state": "0",
"snmpv3_contextname": "",
"evaltype": "0",
"jmx_endpoint": "",
"master_itemid": "0",
"timeout": "3s",
"url": "",
"query_fields": [],
"posts": "",
"status_codes": "200",
"follow_redirects": "1",
"post_type": "0",
"http_proxy": "",
"headers": [],
"retrieve_mode": "0",
"request_method": "0",
"output_format": "0",
"ssl_cert_file": "",
"ssl_key_file": "",
"ssl_key_password": "",
"verify_peer": "0",
"verify_host": "0",
"allow_traps": "0",
"lastclock": "0",
"lastns": "0",
"lastvalue": "0",
```

```
    "prevvalue": "0"
  },
  {
    "itemid": "10010",
    "type": "0",
    "snmp_community": "",
    "snmp_oid": "",
    "hostid": "10001",
    "name": "Processor load (1 min average per core)",
    "key_": "system.cpu.load[percpu,avg1]",
    "delay": "1m",
    "history": "1w",
    "trends": "365d",
    "status": "0",
    "value_type": "0",
    "trapper_hosts": "",
    "units": "",
    "snmpv3_securityname": "",
    "snmpv3_securitylevel": "0",
    "snmpv3_authpassphrase": "",
    "snmpv3_privpassphrase": "",
    "formula": "",
    "error": "",
    "lastlogsize": "0",
    "logtimefmt": "",
    "templateid": "0",
    "valuemapid": "0",
    "params": "",
    "ipmi_sensor": "",
    "authtype": "0",
    "username": "",
    "password": "",
    "publickey": "",
    "privatekey": "",
    "mtime": "0",
    "flags": "0",
    "interfaceid": "0",
    "port": "",
    "description": "The processor load is calculated as system CPU
load divided by number of CPU cores.",
    "inventory_link": "0",
    "lifetime": "0",
    "snmpv3_authprotocol": "0",
    "snmpv3_privprotocol": "0",
    "state": "0",
    "snmpv3_contextname": "",
    "evaltype": "0",
    "jmx_endpoint": "",
    "master_itemid": "0",
    "timeout": "3s",
    "url": "",
```

```
    "query_fields": [],
    "posts": "",
    "status_codes": "200",
    "follow_redirects": "1",
    "post_type": "0",
    "http_proxy": "",
    "headers": [],
    "retrieve_mode": "0",
    "request_method": "0",
    "output_format": "0",
    "ssl_cert_file": "",
    "ssl_key_file": "",
    "ssl_key_password": "",
    "verify_peer": "0",
    "verify_host": "0",
    "allow_traps": "0",
    "lastclock": "0",
    "lastns": "0",
    "lastvalue": "0",
    "prevvalue": "0"
  },
  "id": 1
}
```

## Finding dependent item

Find one Dependent item for item with ID "25545".

Request:

```
{
  "jsonrpc": "2.0",
  "method": "item.get",
  "params": {
    "output": "extend",
    "filter": {
      "type": "18",
      "master_itemid": "25545"
    },
    "limit": "1"
  },
  "auth": "038e1d7b1735c6a5436ee9eae095879e",
  "id": 1
}
```

Response:

```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "itemid": "25547",
      "type": "18",
      "snmp_community": "",
      "snmp_oid": "",
      "hostid": "10116",
      "name": "Seconds",
      "key_": "apache.status.uptime.seconds",
      "delay": "0",
      "history": "90d",
      "trends": "365d",
      "status": "0",
      "value_type": "3",
      "trapper_hosts": "",
      "units": "",
      "snmpv3_securityname": "",
      "snmpv3_securitylevel": "0",
      "snmpv3_authpassphrase": "",
      "snmpv3_privpassphrase": "",
      "formula": "",
      "error": "",
      "lastlogsize": "0",
      "logtimefmt": "",
      "templateid": "0",
      "valuemapid": "0",
      "params": "",
      "ipmi_sensor": "",
      "authtype": "0",
      "username": "",
      "password": "",
      "publickey": "",
      "privatekey": "",
      "mtime": "0",
      "flags": "0",
      "interfaceid": "0",
      "port": "",
      "description": "",
      "inventory_link": "0",
      "lifetime": "30d",
      "snmpv3_authprotocol": "0",
      "snmpv3_privprotocol": "0",
      "state": "0",
      "snmpv3_contextname": "",
      "evaltype": "0",
      "master_itemid": "25545",
      "jmx_endpoint": "",
      "master_itemid": "0",
      "timeout": "3s",
    }
  ]
}
```

```
    "url": "",
    "query_fields": [],
    "posts": "",
    "status_codes": "200",
    "follow_redirects": "1",
    "post_type": "0",
    "http_proxy": "",
    "headers": [],
    "retrieve_mode": "0",
    "request_method": "0",
    "output_format": "0",
    "ssl_cert_file": "",
    "ssl_key_file": "",
    "ssl_key_password": "",
    "verify_peer": "0",
    "verify_host": "0",
    "allow_traps": "0",
    "lastclock": "0",
    "lastns": "0",
    "lastvalue": "0",
    "prevvalue": "0"
  }
],
"id": 1
}
```

## Find HTTP agent item prototype

Find HTTP agent item prototype with request method HEAD for specific host id.

Request:

```
{
  "jsonrpc": "2.0",
  "method": "itemprototype.get",
  "params": {
    "hostids": "10254",
    "filter": {
      "type": "19",
      "request_method": "3"
    }
  },
  "id": 17,
  "auth": "d678e0b85688ce578ff061bd29a20d3b"
}
```

Response:



```
{
  "jsonrpc": "2.0",
  "result": [
    {
      "itemid": "28257",
      "type": "19",
      "snmp_community": "",
      "snmp_oid": "",
      "hostid": "10254",
      "name": "discovered",
      "key_": "item[#{INAME}]",
      "delay": "#{IUPDATE}",
      "history": "90d",
      "trends": "30d",
      "status": "0",
      "value_type": "3",
      "trapper_hosts": "",
      "units": "",
      "snmpv3_securityname": "",
      "snmpv3_securitylevel": "0",
      "snmpv3_authpassphrase": "",
      "snmpv3_privpassphrase": "",
      "formula": "",
      "error": "",
      "lastlogsize": "0",
      "logtimefmt": "",
      "templateid": "28255",
      "valuemapid": "0",
      "params": "",
      "ipmi_sensor": "",
      "authtype": "0",
      "username": "",
      "password": "",
      "publickey": "",
      "privatekey": "",
      "mtime": "0",
      "flags": "2",
      "interfaceid": "2",
      "port": "",
      "description": "",
      "inventory_link": "0",
      "lifetime": "30d",
      "snmpv3_authprotocol": "0",
      "snmpv3_privprotocol": "0",
      "state": "0",
      "snmpv3_contextname": "",
      "evaltype": "0",
      "jmx_endpoint": "",
      "master_itemid": "0",
      "timeout": "3s",
      "url": "#{IURL}",
    }
  ]
}
```

```
"query_fields": [],  
"posts": "",  
"status_codes": "",  
"follow_redirects": "0",  
"post_type": "0",  
"http_proxy": "",  
"headers": [],  
"retrieve_mode": "0",  
"request_method": "3",  
"output_format": "0",  
"ssl_cert_file": "",  
"ssl_key_file": "",  
"ssl_key_password": "",  
"verify_peer": "0",  
"verify_host": "0",  
"allow_traps": "0"  
}  
],  
"id": 17  
}
```

## See also

- [Application](#)
- [Host](#)
- [Graph prototype](#)
- [Trigger prototype](#)

## Source

CItemPrototype::get() in *frontends/php/include/classes/api/services/CItemPrototype.php*.

From:

<https://www.zabbix.com/documentation/4.0/> - **Zabbix Documentation 4.0**

Permanent link:

<https://www.zabbix.com/documentation/4.0/manual/api/reference/itemprototype/get>

Last update: **2019/01/28 14:50**

