

Masterpiece Solutions

# Masterpiece Manager REST API

v2

CONFIDENTIAL

## Table of Contents

Overview of REST API for Masterpiece Manager Version 11 .....	2
Artists .....	3
Items .....	6
Editions .....	8
Categories .....	9
Mediums .....	10
Attributes .....	11
Events.....	13
Company .....	14

## Overview of REST API for Masterpiece Manager Version 11

This only applies to the online version (v11) of Masterpiece Manager.

All calls are “restful”; that is, they are http addresses with options specified as url parameters. Additionally, all the calls use the GET protocol (even if “logically” a PUT, POST or DELETE), for simplicity and consistency.

We have attempted to make them simple, consistent and intuitive to use.

Where applicable, pagination and partial response is supported so only the required data is retrieved.

The general format is <resource>?<param1>=value1&<param2>=value2...

For example, artists?key=123xyz&fields=artistId, lastName, firstName

All calls require a unique “key” that enables retrieval of data for a specific client.

Keys are generated in Administration->Company->Developer Settings.

Note: if the key contains a plus sign (+), the key will have to be modified [by us] before you can use it.

If no other parameters are included, then pagination defaults (for offset and limit) may be applied.

Some calls have optional “filter” parameters; e.g., to limit results to items in a given category.

The same API is used for retrieval and for searching. Searches are accomplished by using the “q” url parameter and fields can be searched by specifying one or more field|value pairs.

All data is sent and returned in JSON (JavaScript Object Notation) format.

Data is returned in string format, or data can be null.

Some numeric data may be in string format, which is not an issue for “loosely typed” PHP but other languages might need to convert.

All data names, whether as url parameters or in returned results, use lowerCamelCase capitalization, which is consistent with javascript conventions and thus fits well with JSON-formatted result sets.

Basically this means single words are lowercase and multiple words are concatenated with no special characters (no underlines or dashes) and all but the first word are capitalized.

Examples: artistId, lastName, title, scancode, createDate

Where applicable, elements are only returned if Display Web is checked (e.g., items, events).

You can test out various parameters easily using a web browser.

Current API is at <https://masterpiece-api.com/v2>

## Artists

<https://masterpiece-api.com/v2/artists/?key=<yourKey>>

Artist data is returned as a JSON object with “total” results and an “artists” array.

The total always equals the entire result set, not the number of items returned on a given call.

E.g., there might be 300 artists, but perhaps only 12 retrieved at a time; total will be set to 300 in this case.

artistId	
biography	
hasBio	0/1
picture	Artist photo
lastName	
firstName	
email	
phone	
line1	
line2	
line3	
city	
state	
zip	
country	
birthDate	yyyy-mm-dd hh:mm:ss
deathDate	yyyy-mm-dd hh:mm:ss
createDate	yyyy-mm-dd hh:mm:ss
itemId	Id of web primary item, if assigned
itemPicture	Web primary item picture, if assigned If not assigned, then we get one at random, UNLESS category is specified, in which case a random pic from that category is used.
description	

*(artists continued)*Optional Parameters

artistId	
category	
medium	
locationId	
fields	Specify what fields to return, using the field names from the artists table. E.g., fields=artistId,lastName,firstName
attributes	One or more attribute names with optional values and optional operator (= or !=) If using values then use   to separate name value. E.g., attributes=gift,metal silver, text1 exclude !=
attributeCompare	Only applies if attributes parameter is used. Defaults to logical OR if more than one attribute specified, but can set attributeCompare=AND
limit	How many items to return. Default is 12.
offset	Where in result set to start returned items. Default is 0. limit and offset can be used for pagination.
q	For searching one or more fields using field value pairs. E.g., q=firstName Sam All searches use SQL LIKE comparisons, and won't make sense for all fields.
qCompare	Default is OR, can set to AND, for use with q
compare	Compare one or more fields using field value operator triplets where operator can be = (equal), != (not equal) > (greater than) or < (less than). E.g., compare=lastName s >,lastName t < for artists with last name starting with S.
emptyNames	emptyNames=1 to include artists with no name
sortOptions	Default is by lastName, firstName. Can specify one or more fields, each with optional direction (as field direction pair), use asc and desc. E.g., sortOptions=createDate desc, lastName

*(artists continued)*

locationId, category, medium and attributes filters are applied to items in the artist's inventory, so for example, using a filter of category=Jewelry would result in only artists that have items with category of Jewelry.

If searching for artists in multiple categories, use the q parameter, e.g.  
q=category|painting,category|sculpture

Attributes optional operator defaults to = and can apply to having the attribute or to the attribute value (see examples in Items API).

## Items

<https://masterpiece-api.com/v2/items/?key=<yourKey>>

Item data is returned as a JSON object with “total” results and an “items” array.

The total always equals the entire result set, not the number of items returned on a given call.

E.g., there might be 300 titles in a specified category, but perhaps only 12 retrieved at a time; total will be set to 300 in this case.

title	
scancode	
description	
lastName	Artist name
firstName	“
category	
medium	
dealerPrice	
retailPrice	
framePriceIncluded	0/1
subject	
circa	year
height	
width	
depth	
weight	
dimUnits	0=English, 1=Metric
frameWidth	
frameHeight	
artistId	
createDate	yyyy-mm-dd hh:mm:ss
itemId	
editionType	0=unique, 1=open, 2=limited, 3=non stock
editionSize	
is3d	0/1
stockLocation	
locationId	
qoh	quantity on hand
notes	
attributes	Array of zero or more attribute elements, each containing: attribute, price, value
pictures	Array of zero or more picture elements, each with full path to the image

*(Items continued)*Optional Parameters

itemId	Can have multiple comma-separated item ID's
artistId	
eventId	Items associated with given event
locationId	
category	
medium	
fields	Specify what fields to return, using the field names from the Items table. E.g., fields=artistId,itemId,title,pictures
attributes	One or more attribute names with optional values and optional operator (= or !=) If using values then use   to separate name value. E.g., attributes=gift,metal silver, text1 exclude !=
attributeCompare	Only applies if attributes parameter is used. Defaults to logical OR if more than one attribute specified, but can set attributeCompare=AND
limit	How many items to return. Default is 12.
offset	Where in result set to start returned items. Default is 0. limit and offset can be used for pagination.
q	For searching one or more fields using field value pairs. E.g., q=scancode ABECAR000004 All searches use SQL LIKE comparisons .
qCompare	Default is OR, can set to AND, for use with q
compare	Compare one or more fields using field value operator triplets where operator can be = (equal), != (not equal), > (greater than) or < (less than). E.g., compare=retailPrice 1000 < for items with price less than \$1000.
dateRestrict	Can filter by some number of days, months, weeks or years, using the createDate. Format is single letter (d, m, w or y) followed by number. E.g., dateRestrict=d30 for last 30 days
sortOptions	Default is by title (ascending). Can specify one or more fields, each with optional direction (as field direction pair), use asc and desc. E.g., sortOptions=createDate desc, title Get random records with sortOptions=RAND()

attributes param can be used to check for the attribute NOT being present, e.g. attributes=text1|!= would return all items that do not have an attribute named text1. It can also be used to check for the attribute NOT having a specific value, e.g. attributes=text1|@exclude@|!= returns items with text1 attributes that do not have @exclude@ as the value. artistDescription not returned but can be queried.



## Editions

<https://masterpiece-api.com/v2/editions/?key=<yourKey>>

Edition data is returned as a JSON object with “total” results and an “editions” array.

editionId	
itemId	Associated Item
edition	#
sold	0/1

### Optional Parameters

editionId	
itemId	
limit	How many items to return. Default is 12.
offset	Where in result set to start returned items. Default is 0. limit and offset can be used for pagination.
fields	Specify what fields to return. E.g., fields=edition,sold
q	For searching one or more fields using field value pairs. E.g., q=sold 0 All searches use SQL LIKE comparisons , and won't make sense for all fields.
sortOptions	Default is by edition. Can specify one or more fields, each with optional direction (as field direction pair), use asc and desc. E.g., sortOptions=editionId

## Categories

<https://masterpiece-api.com/v2/categories/?key=<yourKey>>

Category data is returned as a JSON object with “total” results and a “categories” array.

Unless the fields parameter is used, and mediums not specified, each category has an array of mediums for which it is the parent. This is the default if only the key parameter is sent.

category	
categoryId	
mediums	Array of zero or more medium data
medium	
mediumId	

### Optional Parameters

categoryId	
category	
medium	
artistId	
locationId	
fields	Specify what fields to return, using the field names from the categories table. If mediums specified, you get everything anyway. E.g., fields=category
q	For searching one or more fields using field value pairs. E.g., q=medium Oil All searches use SQL LIKE comparisons , and won't make sense for all fields.
sortOptions	Default is by category.

## Mediums

<https://masterpiece-api.com/v2/mediums/?key=<yourKey>>

Medium data is returned as a JSON object with “total” results and a “mediums” array.

medium	
mediumId	
parentId	The id of the parent category

### Optional Parameters

categoryId	
category	
mediumId	
artistId	
locationId	
fields	Specify what fields to return. E.g., fields=medium
q	For searching one or more fields using field value pairs. E.g., q=category Jewelry All searches use SQL LIKE comparisons , and won't make sense for all fields.
sortOptions	Default is by medium.

## Attributes

<https://masterpiece-api.com/v2/attributes/?key=<yourKey>>

Attribute data is returned as a JSON object with “total” results and an “attributes” array.

attribute	
price	
value	

In most cases the artists or items API will be more applicable.

This is a special case where just attributes, or most likely a specific attribute designated, are needed.

For example, we want all possible values of an attribute called Holidays, and various items have “Christmas”, “Fourth Of July”, “Halloween” etc. as a value for that attribute.

We could then call `attributes?key=<somekey>&attributes=Holiday`

*(attributes continued)*Optional Parameters

attributes	One or more attribute names with optional values. If using values then use   to separate name value. E.g., attributes=gift,metal silver
attributeCompare	Only applies if attributes parameter is used. Defaults to logical OR if more than one attribute specified, but can set attributeCompare=AND
locationId	
limit	How many items to return. Default is 12.
offset	Where in result set to start returned items. Default is 0. limit and offset can be used for pagination.
q	For searching one or more fields using field value pairs. E.g., q=value sale All searches use SQL LIKE comparisons , and won't make sense for all fields.
compare	Compare one or more fields using field value operator triplets where operator can be = (equal), > (greater than) or < (less than). E.g., compare=price 1000 < for items with attribute price less than \$1000.
sortOptions	Default is by attribute, value.

## Events

<https://masterpiece-api.com/v2/events/?key=<yourKey>>

Event data is returned as a JSON object with “total” results and an “events” array.

eventId	
title	
startDate	
endDate	
body	Rich text blob
itemId	Id of event item, if assigned
itemPicture	Event item picture, if assigned
locationId	

### Optional Parameters

eventId	
eventWhen	current, future or past current means start or end date is this week
locationId	filter by location
limit	How many items to return. Default is 12.
offset	Where in result set to start returned items. Default is 0. limit and offset can be used for pagination.
fields	Specify what fields to return. E.g., fields=eventId, title, startDate, endDate
q	For searching one or more fields using field value pairs. E.g., q=startDate 2012 All searches use SQL LIKE comparisons , and won't make sense for all fields.
sortOptions	Default is by startDate descending. Can specify one or more fields, each with optional direction (as field direction pair), use asc and desc. E.g., sortOptions=endDate asc, title

## Company

<https://masterpiece-api.com/v2/company/?key=<yourKey>>

Company data is returned as a JSON object with company data, an “info” array and a “locations” array.

name	
description	
termsAndConditions	
logo	
fromEmail	
fromName	
locations	Array of location data
locationId	
locationName	
email	
phone	
address	
city	
state	
zip	
country	
hoursOfOperation	
info	Array of info data
infoId	
infoName	
info	Up to 4000 chars

The info data is labeled ‘More Info’ on the General tab of the Administration Company page.

### Optional Parameters

locationId	
locationName	