

Bridge : Postal Classes and Function List

WUPQueryInfo

- **void ExcludeInt(int id)**
 - Add and int to the list of ints to ignore (use is subclass dependent)
- **void ExcludeInts(int[] ids)**
 - Add multiple ints to the list of ints to ignore (use is subclass dependent)
- **void AddInt(EWUPCategoryMethod method, int value)**
 - Stores an int to search for (use is subclass dependent)
- **void AddInts(EWUPCategoryMethod method, int[] values)**
 - Stores multiple ints to search for (use is subclass dependent)
- **void SetNiceName(string slug)** (alias for SetSlug)
- **void SetSlug(string slug)**
 - Specify exactly one slug. Clears any existing slugs
- **void AddSlug(EWUPCategoryMethod method, string value)**
 - Stores a slug to search for (use is subclass dependent)
- **void AddSlugs(EWUPCategoryMethod method, string[] values)**
 - Stores multiple slugs to search for (use is subclass dependent)

PostBasics (WUPQueryInfoBasics)

INHERITS WUPQueryInfo

- **void SpecifyPostId(int id)**
Find a specific post by it's unique id
- **void SpecifyPageId(int id)**
Find a specific page by it's unique id
- **void SpecifyCommentCount(int count, string count_operator = "=")**
Specify how many or "at least how many" comments a post must have before it gets returned
- **void SpecifyPostPassword(string password)**
If the post is password protected, what is the password?
- **void SpecifyPasswordRequired(bool required)**
Specify whether or not the results must be password protected or not or if it doesn't matter
- **void SpecifyPermission(string perm)**
If you want to limit search functionality to users with a specific capability, specify the capability
- **void SpecifySearchTerm(string keyword)**
Search for posts containing a custom string (Like a search bar on a website)
- **void SpecifyTitle(string title)**
Find a specific post by it's slug

- **void SpecifyPostType(string slug)**
Specify the custom post type you are searching for, or just the normal page or post type
- **void SpecifyPostTypes(string[] slugs)**
Specify a collection of post types to query / return
- **void SpecifyPostNameIn(string post)**
Add an additional slug to search for. Results will contain at least one of the specified slugs
- **void SpecifyPostNamesIn(string[] posts)**
Specify multiple (additional) slugs to search for. Results will contain at least one of the specified slugs
- **void SpecifyPostStatus(EWUPostalStatus status)**
Specify the required status of valid results. I.e. published, draft etc.
- **void SpecifyPostStatuses(EWUPostalStatus[] statuses)**
Specify multiple allowed post statuses
- **void SpecifyPageName(string name)**
- **void SpecifyPostName(string name)**
- **void SpecifyPostIn(int post)**
- **void SpecifyPostsIn(int[] post)**
- **void SpecifyPostParent(int parent)**
- **void SpecifyPostParentIn(int parent)**
- **void SpecifyPostParentsIn(int[] parent)**
- **void ExcludePost(int post)**
- **void ExcludePosts(int[] post)**
- **void ExcludePostParent(int parent)**
- **void ExcludePostParents(int[] parent)**
- **void SpecifyOrderBy(EWUPOrderBy name, EWUPOrder order)**

Author (WUPQueryInfoAuthor)

INHERITS WUPQueryInfo

- **void AddId(int id)**
- **void AddIds(int[] ids)**
- **void AddIdIn(int id)**
- **void AddIdsIn(int[] ids)**
- **void Exclude(int id)**
- **void ExcludeMultiple(int[] ids)**

Category (WUPQueryInfoCategory)

INHERITS WUPQueryInfo

- **void AddSlug(string slug)**
- **void AddSlugs(string [] slug)**
- **void AddId(int id)**
- **void AddIdIn(int id)**
- **void AddIdAnd(int id)**
- **void AddIds(int [] ids)**
- **void AddIdsIn(int [] ids)**
- **void AddIdsAnd(int [] ids)**
- **void Exclude(int id)**
- **void ExcludeNot(int id)**

- void ExcludeMultiple(int [] ids)
- void ExcludeMultipleNot(int [] ids)

Tags (WUPQueryInfoTags)

INHERITS WUPQueryInfo

- void AddSlug(string slug)
- void AddSlugs(string [] slug)
- void AddSlugIn(string slug)
- void AddSlugsIn(string [] slug)
- void AddSlugAnd(string slug)
- void AddSlugsAnd(string [] slug)
- void AddId(int id)
- void AddIdIn(int id)
- void AddIdAnd(int id)
- void AddIds(int [] ids)
- void AddIdsIn(int [] ids)
- void AddIdsAnd(int [] ids)
- void Exclude(int id)
- void ExcludeMultiple(int [] ids)
- void ExcludeMultipleNot(int [] ids)

WUPTaxQueryEntry

Members

- WUPNestedQueryArray<WUPTaxQueryEntry> QueriesArray

Methods

- void AddTerms(List<int> values)
- void AddTerms(List<string> values)
- WUPTaxQueryArray AddTermsArray()

WUPNestedQueryArray<WUPTaxQueryEntry>

Members

- List<WUPTaxQueryEntry> Entries
- WUPNestedQueryArray<WUPTaxQueryEntry> NestedLevel
- int EntriesCount

Methods

- AddEntry(WUPTaxQueryEntry value, EWUPAndOr relation)
- AddEntry(WUPTaxQueryEntry value)
- WUPNestedQueryArray<T> CreateNestedEntries()

Taxonomy (WUPTaxQuery)

Members

- WUPTaxQueryArray Entries

WUPDateQueryElement

Members

- public int? Year
- public EWUPDateMonths? Month
- public int? Week
- public EWUPDateDays? Day
- public int? Hour

- public int? Minute
- public int? Second
- public EWUPDateCompareOperator? Compare
- public string Column
- public EWUPAndOr? Relation
- public bool? inclusive

Methods

- void SetBeforeDate(int year, EWUPDateMonths month, EWUPDateDays day)
- void SetAfterDate(int year, EWUPDateMonths month, EWUPDateDays day)
- void SetDaysOfWeek(int[] daysFrom0To6)

WPQueryInfoDate

Members

- public int? Year
- public EWUPDateMonths? Month
- public int? Week
- public EWUPDateDays? Day
- public int? Hour
- public int? Minute
- public int? Second

Methods

- void SpecifyYearMonth(int year, EWUPDateMonths month)
- void SpecifyYearMonth(int year_and_month)
- void SpecifyDate(int? y, EWUPDateMonths? m, int? w, EWUPDateDays? d, int? hour, int? minutes, int? seconds)
- WUPDateQueryElement AddDateQuery(int? y, EWUPDateMonths? m, int? w, EWUPDateDays? d, int? hour, int? minutes, int? seconds)

Pagination (WUPPagination)

- void SpecifyPaging(bool use = true)
 - Do you want to limit results to a specific number of posts or return all?
- void IgnoreStickyPosts(bool ignore = true)
 - Return sticky posts also or skip them?
- void SpecifyPostsPerPage(int count)
 - If using paging, max posts to return per query
- void SpecifyPostsPerArchivePage(int count)
 - number of posts to show per page – on archive pages only. Overrides posts_per_page and showposts on pages
- void SpecifyOffset(int count)
 - Number of posts to skip (**NOTE:** Breaks paging)
- void SpecifyPage(int count)
 - number of page for a static front page. Show the posts that would normally show up just on page X of a Static Front Page
- void SpecifyPaged(int count)

number of page. Show the posts that would normally show up just on page X when using the “Older Entries” link

Caching (WUPCaching)

- `void SetCacheResults(bool use = true)`
- `void SetUpdatePostMetaCache(bool update = true)`
- `void SetUpdatePostTermCache(bool update = true)`

MimeTypes (WUPMimeTypes)

- `void SpecifyType(string mime)`
- `void SpecifyTypes(string [] mimes)`