ê³ ê¸‰ ë“±ê¸‰Â íŽ˜ì´ì§€

°œìš”

ì´ íŽ˜ì´ì§€ë¥¼ ì‚¬ìš©í•˜ì—¬ íŠ¸ëž˜ì»¤ í•ëª© í˜¹ì€ ìœ„í‚¤ íŽ˜ì´ì§€ë¥¼ í‰ê°€í•˜ê¸° ìœ„í•œ "ë“±ê¸‰" ì‹œìŠ¤í…œì„ êµ¬ì„±í•˜ì‹œì˜¤.

'ê·¼í•˜ë ¤ë©´

ê´€ë¦¬ íŒ¨ë„ ìƒì˜ **ë“±ê¸‰** ì•„ì´ì½˜ ì„ í´ë¦í•˜ì‹œì˜¤

ë˜ëŠ”

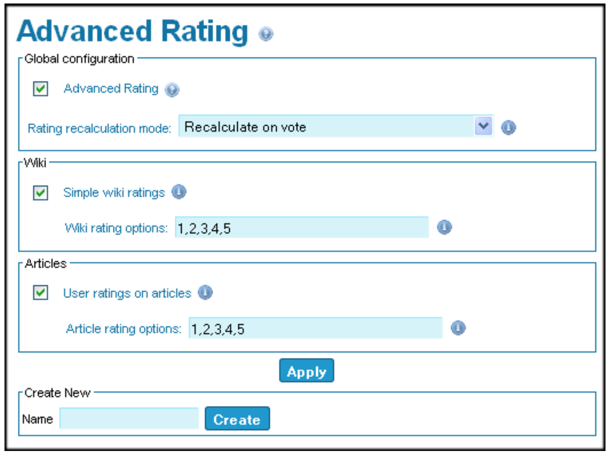
http://ê·€í•˜ë„ë©”ì¸ì£¼ì†Œ.com/tiki-admin.php?page=rating ìœ¼ë¡œ ì 'ê·¼í•˜ì‹œì˜¤

¼ì˜

í‹°í‚¤ëŠ” í˜„ìž¬ ê³ ê¸‰ ë“±ê¸‰ì„ íˆµí•˜ì—¬ ë¶„ë¥˜í•˜ëŠ” ê²ƒì„ ë‹¤ìŒì˜ ê¸°ëŠ¥ë“¤ì—ì„œ ì§€ì›í•©ë‹ˆë‹¤:

- ê¸°ì‚¬
- ìœ„í‚¤
- ëŒ“ê¸€

í•œ ì˜ˆì œ ì—°ì‚° íŠ¸ëž˜ì»¤ í•„ë“œ ë¥¼ ì°¸ì¡°í•˜ì‹œì˜¤



ê³ ê¸‰ ë“±ê¸‰ íŽ˜ì´ì§€

ê´€ë ¨ í† í”½

- ì „ìž ë¯¼ì£¼ì£¼ì˜ ì‹œìŠ¤í…œ
- ë²„ê·¸ ë° í¬ë§·ì‚¬í•
- ë“±ê¸‰

ì„¤ì •	ì„¤ëª…	Default
ì „ì— êµ¬ì„±		
ê³ ê¸‰ ë“±ê¸‰	ë‚´ë¶€ ë“±ê¸‰ ì‹œìŠ¤í…œì„ í™œì„±í™”í•˜ì—¬ ì´ëŠ” íŠ¸ëž˜ì»¤, ê¸°ì‚¬ í˜¹ì€ ê¸°íƒ€ ê¸°ëŠ¥ì—ì„œì˜ ê°'ë“¤ì„ ê³„ì‚°í•˜ëŠ”ë° ì‚¬ìš©ë¨.	
ë“±ê¸‰ ìž¬ê³„ì‚° ëª¨ë“œ:	ë“±ê¸‰ í•©ê³„ê°€ ì–¸ì œ ê·¸ë¦¬ê³ ì–´ë–»ê²Œ ìž¬ê³„ì‚°ë˜ëŠ"ì§€ë¥¼ ê²°ì •: * **íˆ¬í'œ ì‹œ** (ê¸°ë³¸ê°'): íˆ¬í'œê°€ ì´ˆí–‰ë ë•Œ ë§¤ë²ˆ ë§Œì ê¸°ì¤€ì— ëŒ€í•œ ì ìˆ˜ê°€ ìž¬ê³„ì‚°ë˜ì–´ì•¼í•¨ì„ ì§€ì •. ì´ ì„ íƒì€ ê·œëª¨ê°€ ë" ìž‘ì€ ì‚¬ì´íŠ¸ ë° ë"±ê¸‰ì´ ì‚¬ìš©ë ë•Œ ë¹„êµì ë‹¨ìˆœí•œ ì—°ì‚° ë°©ì‹ì— ì ìš©í•¨. * **ë¡œë"œ ì‹œ ë¬´ìž'ìœ„ë¡œ**: ì´ëŠ" ë¬´ìž'ìœ„ ê¸°ë°˜ìœ¼ë¡œ íŽ˜ì´ì§€ ë¡œë"œ ì‹œì— ì¼ë¶€ ì†Œìˆ˜ì˜ ì ìˆ˜ë"¤ì´ ê³„ì‚°ë˜ë„ë¡ will cause a few scores to be calculates on page load on a random basis (odds and count can be configured to adapt to site load). This option is suitable for calculation rules involving time that must be recalculated even if no new votes occurred. * **Random on vote** is similar to random on load, but will recalculate multiple scores (not necessarily including the current object) when a vote is performed. It is suitable for similar situations. The best option will depend on site load. * **Periodic**: is the best option for heavy load sites, making sure all calculations are done outside the web requests. A cron job must be set-up manually by the site's administrator. A sample script is available at the end of this page. Depending on the site load, some options may be better than others; on large volume sites, we recommend **cron job**. The **Recalculate on vote** recalculation may be inaccurate if rating calculation depends time. **Before any attempt to re-index the object:** Ties into the Search and List from Unified Index and updates the calculation at index-time.	Recalculate on vote
Recalculation odds (1 in X):		
Recalculation count:		
Wiki		
Simple wiki ratings	Enable a simple rating bar at the top of each wiki page.	
Wiki rating options:	List of options for the simple wiki ratings.	1,2,3,4,5

ì„¤ì •	ì„¤ëª…	Default
Articles	Enable a simple rating bar at the top of each articles page.	
User ratings on articles		
Article rating options:		

he feature must first be enabled through this same administration panel. Along with the feature, a few options are available. Among them, the score ecalculation period must be defined. These are the available options:

- **On vote (default)** indicates that the score for the object should be recalculated every time a vote is performed. This option is suitable for sites with lower volumes and relatively simple calculation methods when ratings are used.
- **Random on load** will cause a few scores to be calculates on page load on a random basis (odds and count can be configured to adapt to site load). This option is suitable for calculation rules involving time that must be recalculated even if no new votes occurred.
- **Random on vote** is similar to random on load, but will recalculate multiple scores (not necessarily including the current object) when a vote is performed. It is suitable for similar situations. The best option will depend on site load.
- **Periodic** is the best option for heavy load sites, making sure all calculations are done outside the web requests. A cron job must be set-up manually by the site's administrator. A sample script is available at the end of this page.

or the random options, the odds of recalculating must be specified as a dice roll. For each occurrence of a recalculation, a limit to how many scores can e calculated must be specified to avoid the hang-up effect on the page load.

he value ranges for each object type can also be specified through the administration panels.

he common *sort_mode* parameter to lists can be used to activate sorting using advanced ratings. To do so, the sort mode must be set to *dv_rating_X_asc* or *adv_rating_X_desc* where *X* is the ID of the rating configuration. The default sort can also be set to advanced ratings in the dministration panel where applicable.

Calculation configuration

rom the administration panel, new calculations can be added. Initially, only the name is required. When created, the calculation will contain suitable efault values.

or wiki pages:

nus, visitors can provide feedback like:

- Did this page help you solve the issue?
- Was this page easy to understand?

Advanced Rating

troduced in Tiki5, the advanced rating feature allows for more control over the aggregation of scores.

ating methods are defined globally and will be used for all supported objects. They are defined through the **Advanced Rating** administration panel (tiki dmin.php?page=rating). Multiple methods can be created. If a method contains type-specific calculations, it will be ignored when performing the alculation.

eatures currently supporting sorting through advanced rating:

- Articles
- Wiki
- Comments

Sorting items according to advancedÂ rating

ote that the sort mode to use when needing to sort by advanced rating is either adv_rating_xx_asc or adv_rating_xx_desc, where xx is the ratingConfigId

eature request: Can we make this take the name of the config instead of the ratingConfigId as well?

Set-up

y default, each calculated value is kept for 1 hour (3600 seconds). This limit does not apply when recalculating on vote, but is used for every other echnique to avoid recalculating the same scores over and over again.

he calculation is defined as a small piece of code, similar to functional languages, which is very close to mathematical representations. Creating custom ormulas is expected to require some mathematical skills. However, this documentation should provide examples for most frequent cases.

he editor in the administration panel performs extensive validation and will make it impossible to save the formula unless it can be evaluated. Checks ar erformed for:

- Syntax errors
- Unknown functions
- Missing arguments
- Invalid argument values
- Unknown input variables

Default formula
(rating-average (object type object-id))

can be altered to limit the vote consideration to a limited time span, 30 days for example.

Recent votes only
(rating-average (object type object-id) (range (mul 3600 24 30)))

the language, spaces do not matter. Only the parenthesis indicate structure. **rating-average** is a function that fetches the ratings for a given object. *ype* and *object-id* are standard variables fed when calculating a rating. **object** and **range** are configuration options of the function.

ul is a mathematical function. (mul 3600 24 30) is equivalent to 3600*24*30.

he functions can be combined in various ways. For example, we could calculate a score that considers the votes from the past month, but gives extra

mphasis on the recent ones.

Combined vote duration

```
(add (rating-average (object type object-id) (range (mul 3600 24 30))) (rating-average (object type object-id) (range (mul 3600 24 7))) )
```

ven though the votes are 1-5, the final score can be on an entirely different scale. The language is also extensible if the calculation needs to be combine
ith other factors or weight. See Rating Language.

ll available options are documented in the following section.

General Reference

comment

ny comment block is stripped from the formula at parse-time

Examples

```
(mul 1 2 (comment Simple enough?)) -> 2
```

mul (Multiply)

erforms a simple multiplication accepting multiple input values.

Examples

```
(mul 3 4) -> 12 (mul (mul 3 4) 5) -> 60 (mul 3 4 5) -> 60 (mul 4 0.5) -> 2
```

div (Divide)

erforms a simple division accepting multiple input values.

Examples

(div 3 4) -> 0.75 (div (mul 3 10) 5) -> 6 (div 30 5 3) -> 2 (div 4 0.5) -> 8

add (Sum)

erforms a simple sum accepting multiple input

Examples

(add 3 4) -> 7 (add (add 3 4) 5) -> 12 (add 3 4 5) -> 12 (add 4 0.5) -> 4.5

sub (Substract)

erforms a simple substraction accepting multiple input

Examples

(sub 3 4) -> -1 (sub (sub 3 4) 5) -> -6 (sub 3 4 5) -> -12 (add 4 0.5) -> 3.5

round

ounds to a specific number of digits (new in Tiki12)

Examples

(round 4.556234342234 2) -> 4.56 (round 4.556234342234) -> 5

coalesce

eturns the first non-empty value from the list.

Examples

(coalesce 3 4) -> -3 (coalesce (sub 3 3) 5) -> 5 (coalesce 0 0 (str) -10) -> -10 (coalesce 0 0 0 0 0) -> 0

str

enerates a static string when needed and the processor attempts to process the string as a variable. Any arguments will be concatenated using spaces.

ote: The quoted string syntax was included in Tiki13.

Examples

(str hello-world) -> "hello-world" (str hello world) -> "hello world" (str hello world foobar) -> "hello world foobar" (str (mul 2 3) "= 6") -> "6 = 6"

concat

oncatenates a string of text. (new in Tiki12)

ote: The quoted string syntax was included in Tiki13.

Examples

(concat (str $) 1234) -> "$1234" (concat 14 (str %)) -> "14%" (concat 14 "%") -> "14%"

map

enerates a map (or dictionary).

Examples

(map (key1 1) (key2 2) (key3 (str value3))) -> {"key1": 1, "key2": 2, "key3": "value3"}

equals

ompares multiple values.

Examples

(equals 2 (add 1 1) (sub 4 2)) -> 1 (equivalent of 2 == 1+1 && 2 == 4-2) (equals (add 1 1) 3) -> 0

f

onditionally evaluates a branch.

Examples

(if (equals 2 2) 42 -1) -> 42 (if (equals 2 1) 42 -1) -> -1

and

nsures all elements evaluate to true.

Examples

(and 3 2 1 2 3) -> 1 (and 2 3 0 2) -> 0

or

nsures that at least one element evaluates to true. Elements are evauated sequentially until a false element is found. Others are left unevaluated.

Examples

(or 3 2 1 2 3) -> 1 (or 2 3 0 2) -> 1 (or 0 0) -> 0

hash

enerates a hash based on multiple values. Used primarily to generate aggregate hashes in the PluginActivityStream. Note that because it is a hash, the xact value coming out does not matter. Only that given the same parameter, it will produce the same value.

Examples

(hash 1) -> [sha1("1")] (hash 1 2 3 4) -> [sha1("1/2/3/4")] (hash 1 2 (map (a 3) (b 4))) -> [sha1("1/2/3/4")]

avg

alculates the average of multiple values. All entries in the list will be flattened if arrays are present.

Examples

(avg 1 2 3) -> 2 ... given list contains [1, 2, 3] (avg list) -> 2

split-list

roduces a multi-dimensional array out of a text string. Each line is expected to be an independent value, each line will be split by a separator into the pecified keys.

Examples

... given str contains a list of 3 comma-separated values (split-list (content str) (separator ,) (keys a b c)) -> [{a: 1, b: 2, c: 3}, {a: 2, b: 3, c: 4}]

for-each

or a list of value pairs, such as the output of *split-list*, evaluates a formula for each set of values, returns the list of results.

Within the formula, variables coming from the list will be used first. Fallback will be on the other variables available in the execution context.

Examples

... given items contains [{a: 1, b: 2, c: 3}, {a: 2, b: 3, c: 4}] (for-each (list items) (formula (mul a b c))) -> [6, 24] ... given items contains [{a: 1, b: 2, c: 3}, {a: 2, b: 3, c: 4}] ... and d contains 10 (for-each (list items) (formula (mul c d))) -> [30, 40]

è³ ê¸‰ ë“±ê¸‰-íŠ¹ì •Â ì°¸ì¡°

rating-average (ë“±ê¸‰-í‰ê·) ë° rating-sum (ë“±ê¸‰Â í•©ê³„)

“±ê¸‰ í•¨ìˆ˜ëŠ” ë“±ê¸‰ ê¸°ë¡ í…Œì´ë¸”ì—ì„œ ì ìˆ˜ë¥¼ ê³„ì‚°í•©ë‹ˆë‹¤. ì‚¬ì´íŠ¸ì— ì ìˆ˜í–‰ëœ ê° ë“±ê¸‰ì€ ëºì´í„°ë² ì´ìŠ¤ ë‚´ì— ë³´ê´€ë˜ë© ì‚¬ìš©ìž €ì • ë“±ê¸‰ì„ ê³„ì‚°í•˜ê¸° ìœ„í•´ ì‚¬ìš©ë ìˆ˜ ìžˆìŠµë‹ˆë‹¤. ë‹¤ì–‘í•œ ì„ íƒì‚¬í•ì´ ë¬¸ì„œì˜ í’ˆì§ˆ í–¥ìƒì„ ì§€ì›í•˜ê³ ë‚˜ í•„ë“œ ìˆ˜ì§‘ê¸°ë¡ ë“¤ì–´ì˜¤ëŠ” ë°ì´í„°ì˜ ìˆœìœ„ë¥¼ ë§¤ê¸°ê¸° ìœ„í•˜ëŠ” ë“±ì˜ ì‚¬ì´íŠ¸ ìƒí˜¸ìž‘ìš©ì„± ë°˜ì˜í•˜ê¸° ìœ„í•´ì—¬ ì ìˆ˜ë¥¼ ê³„ì‚°í•˜ë„ë¡ ì‚¬ìš©ë©ë‹ˆë‹¤.

- **object (ê°œì²´)**, í•„ìˆ˜ì´ë©° í•ìƒ ë‚´ìš©ìž (*ê°œì²´ ìœ í˜• ê°œì²´ID)*.
- **range (ë²”ìœ„)**, to limit how long votes are considered. Argument is provided as a number of seconds.
- **ignore (ë¬´ì‹œ)**, with *anonymous* as an argument to only consider votes from registered users.
- **keep (ìœ ì§€)**, to only consider one vote per visitor. Unless the option is present, all of the votes are taken into account. The option can be either *latest* or *oldest* to indicate which one to keep.
- **revote (ìž¬íˆ¬í‘œ)** can be specified if **keep** is specified. Indicates the time period required between votes. For example, users could be allowed to vote more than once per day, but only their latest vote each day would be considered, if revote is set to mul(24 3600). If the user voted yesterday as well as today, both votes will be counted.

article-info

³„ì‚°ì— í¬í•¨ëœ ì˜ˆ ìžˆëŠ” ê¸°ì‚¬ì—ì„œ ì •ë³´ë¥¼ ì¶”ì¶œ. ì²« ë²ˆì§¸ ì¸ìžëŠ” ì–¸ì œë‚˜ íƒ€ìž… 'article' (ê¸°ì‚¬)ê°€ ë˜ì–´ì•¼í•¨ . ê·¸ ì™¸ ë‹¤ë¥¸ ê°’ì´ ë˜ëŠ” ê²½ìš°, –°ì‚°ì€ í‰ê°€ëœ ê°œì²´ì— ëŒ€í•˜ì—¬ ê±´ë„ˆë›°ì–´ì§€ê²Œ ë˜ì–´, ì´ë¥¼ ê³µì‹ ìœ í˜•-íŠ¹í™”ì‹œìœ¼ë¡œ ë²„ë¦½ë‹ˆë‹¤.

¬ìš©ê°€ëŠ¥í•œ ì†ì„±ë“¤:

- rating (ë“±ê¸‰), ê¸°ì‚¬ì— ìž¨ë¶€ëœ ì •ì ë“±ê¸‰
- view-count
- age-second
- age-hour
- age-day
- age-week
- age-month

ì˜ˆì œ

(article-info type object-id rating) (article-info (str article) 42 age-month)

†ì„±

³´í†µì˜ ê°œì²´ ì†ì„±ì—ì„œ ì •ë³´ë¥¼ ì¶”ì¶œí•¨.

ì˜ˆì œ

(attribute (object type object-id) (property tiki.proposal.accept)) -> [value for page in a rating calculation] (attribute (object (str wiki page) 14) (property tiki.proposal.accept) (default 0)) -> [value for page id 14]

racker-field

¸ëžˆì»¤ í•ëª©ì—ì„œ ë•°ì´í„°ë¥¼ ì¶”ì¶œí•¨. í•„ë“œ ê°’ì€ ì^«ìžëiœ ìžë™ìœ¼ëiœ ë³€í™˜ë¨. ê°’ì´ ë°œê²¬ë˜ì§€ ì•Šê±°ë‚˜ ì šš© ë¶ˆê°€ëŠ¥ í•œ ê²½ìš° 0 ì´ œê³µë¨.

ì˜ˆì œ

(tracker-field (object type object-id) (field priority)) -> [value contained in the tracker item field with permanent name ''priority'']

category-present

°œì²´ìfì— ìi´ìž¬í•˜ëŠ” ëª¨ë“ ë‚˜ì—´ëœ ë²”ì£¼ì˜ ì^˜ì— 1 ì„ ë¶€ì—¬í•¨.

ì˜ˆì œ

(category-present (object type object-id) (list 3 4)) -> [0, 1 or 2 - Depending on how many of categories 3 or 4 are on the object]

ë¶€ëi

µí•© ê²€ìƒ‰ (unified search) ì´ ì‚¬ìš©ë ë•Œ, ìž¬ê³„ì‚°ì€ ìž¬ìƒ‰ì¸ ë™ì•ˆ ì´ë£¨ì–´ì§€ê²Œë„ëi êµ¬ì„±ë ìˆ˜ ìžˆìŠµë‹ˆë‹¤. ê³ ëiœ ì´ ìŠ¤í¬ë¦½íŠ¸ê°€ í•„ìš”í–ˆì§€ë‹ˆë‹¤.

Cron job