

PĀḲI TĪPIṬAKA PROJECTOR - INSTRUCTIONS

version 1.0 (180501)

Table of Contents

Terms of Use.....	1
General Notes.....	1
Content - Bilingual Pāli Text.....	1
Dictionary - Look up & Display Mode.....	2
Dictionary - Dictionary-In-Use Loader Check to Enable.....	2
Content - Loading Data Message.....	3
Content – The Content Browser Menu.....	3
Dictionary - Understanding the List Paragraph Mode.....	4
Dictionary – Fuzzy-[yes] and Dual-[yes].....	4
Dictionary – More On Dual-[yes].....	5
Dictionary – 4 Types of look up Results.....	5
Dictionary – The Weakness of Fuzzy Search.....	6
Dictionary – The Floating Paragraph At The Bottom.....	7
Dictionary – 3 Types of Removing Definition.....	7
Dictionary – Pop-up singleWord Mode.....	8
Software – For Developers.....	9
Tips 1: if web-browser does not support localStorage.....	9
Adding Dictionary Step 1 – Prepare a dictionary array in Javascript.....	10
Step 2 – Add codes to the index files.....	10
2.1 Dictionary-In-Use Loader Check to Enable.....	10
2.2 window.onload = function () {.....	10
Step 3 – Add codes to stlookup_jquery.js file.....	11
3.1 DICTIONARY VARIABLES AND PERFORM LOADING.....	11
3.2 Call look up function for your dictionary here.....	11
3.3 Optional: Font/Styles/Function for Your Specific Dictionary.....	11
Software – Licenses & Credits.....	13
Software – New version & Feedback.....	18

Terms of Use

This software (web-application) is released "as is" with the hope that may it be helpful to users. The developers/contributors of this application take no responsibilities or warranties of anything related to quality or thing of any kind related to the use of this application under any circumstances.

This work does include or use with/without modified source code and/or dictionary data that have been created by others. They may carry one or more types of different licenses.

Please check Software – Licenses & Credits session for more information.

As permissions granted to us to use some of the materials in this work with the conditions that this work is for free distribution only. Thus, please kindly be noted that:

This Pāḷi Tipiṭaka Projector is for free distribution only.

The materials created by others could still have many other rights reserved by their owners. Thus, it is advisable to contact them for more information if you would have any questions related to their materials.

reminder

Please remember that having looked up words in dictionaries alone may not help you understand a Pāḷi sentence correctly. One may need to learn pāḷi properly and acquired other factors to understand it better.

Sometimes, a definition given in a particular dictionary may not be correct. Yes, it happens. You may need to cross-check with other reliable sources/dictionaries sometimes.

Hope that this Pāḷi Tipiṭaka Projector will be useful for you!

Happy reading!

General Notes

Network: This app runs **completely off-line**, no need Internet connection.

Reset: at any time, if you have any troubles with Pāli Tipiṭaka Projector, or just simply want to refresh it, you can reset it by using the file [02_RESET_Tipitaka_Projector.htm](#).

Start: click on the file [01_START_Tipitaka_Projector.htm](#) to run Pāli Tipiṭaka Projector.

Compatible platform: It was tested with **Firefox browser** on Linux OS Computer, Window OS Computer, Android tablet; on iPad 2 (using File Master and its built-in browser).

Pāli Tipiṭaka Projector is a HTML/Javascript web application. Thus it may work on other platforms too if it has a standard web-browser supporting *Javascript* and *localStorage*. However if your device's security setting disabled *Javascript* and *localStorage* then the dictionary function shall not work. Then you may only view the text, no dictionary function.

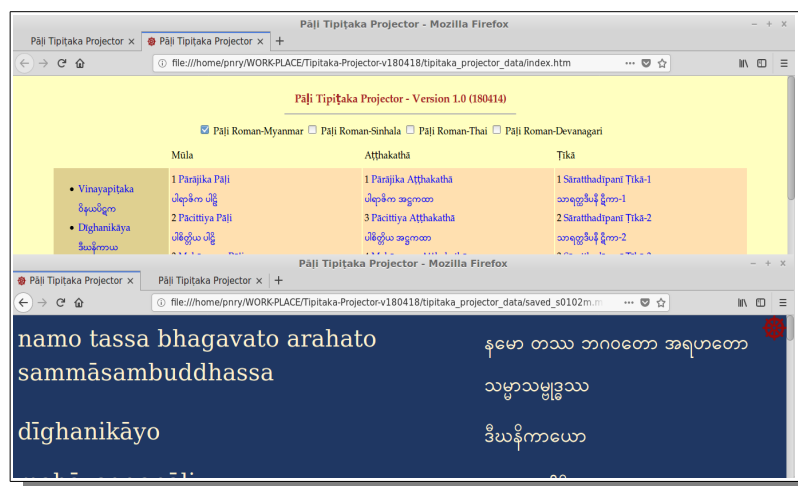
Check Tips 1: if web-browser does not support *localStorage* for a “work-around” solution.

Web-browser: This app is mainly tested with Firefox Browser. Thus, it is advisable using this app with Firefox browser.

Checkbox: ☐ check/un-check to enable/disable it. You can click on the **label** or the checkbox itself to check/un-check it.

Content - Bilingual Pāli Text

Check to convert Pāli Roman into other Pāli characters. Read both of them at the same time.

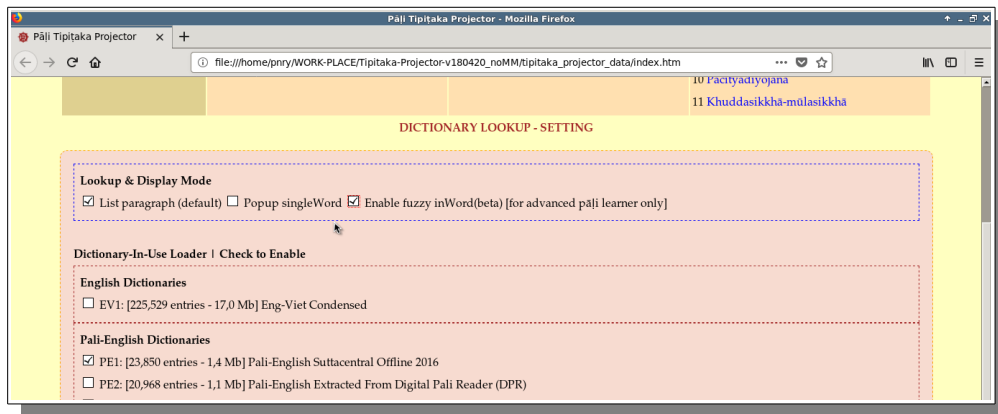


- Pāli Roman-Myanmar: shows Pāli Roman (left) - Pāli Myanmar (right).
- Pāli Roman-Sinhala: shows Pāli Roman (left) - Pāli Sinhala (right).
- Pāli Roman-Thai: shows Pāli Roman (left) - Pāli Thai (right).
- Pāli Roman-Devanagari: shows Pāli Roman (left) - Pāli Devanagari (right).

The converted text shall be (still) in Unicode (UTF-8), however, if your browser can not display the Pāli Myanmar/Sinhala/Thai/Devanagari characters correctly, you may need to install Pāli Myanmar/Sinhala/Thai/Devanagari fonts.

For example: the *ṭ* and *ṇ* character shall not be rendered correctly in Pāli Thai if your device doesn't have the required Pāli Thai fonts.

Dictionary - Look up & Display Mode



- **List paragraph (default):** this mode lists meanings of all words in the paragraph or sentence that you have (just) clicked on it.
- **Pop-up singleWord:** this shows meaning(s) of a single word that you have clicked on.
- **Enable fuzzy inWord(beta):** When it couldn't find any definitions in all of the enabled dictionaries, it will try to split that word (normally it is a compound/long word) into smaller possible portions and find their meanings in the enabled dictionaries.

This function is **not reliable** and should be used for **advanced pāli users only** as it might break wrongly sometimes. Use with caution!

Dictionary - Dictionary-In-Use Loader | Check to Enable

Check/un-check to enable/disable a dictionary.

Important note:

On old-aged devices or not a standard web-browser: if you enable all dictionaries and/or bilingual language function at once, it may cause the web-browser “crashed” when you open Pāli Tipiṭaka Projector. When you cannot open the Pāli Tipiṭaka Projector, please try the following instructions:

If you cannot open or simply want to refresh Pāli Tipiṭaka Projector at any time, you may try opening the file **02_RESET_Tipitaka_Projector.htm** and click on the button **RESET PALI TIPITAKA PROJECTOR** to refresh Pāli Tipiṭaka Projector's Settings to its original state.

Now try opening Pāli Tipiṭaka Projector again. If it was crashing because of loading so many Mb of dictionaries, reduce the number of dictionaries this time.

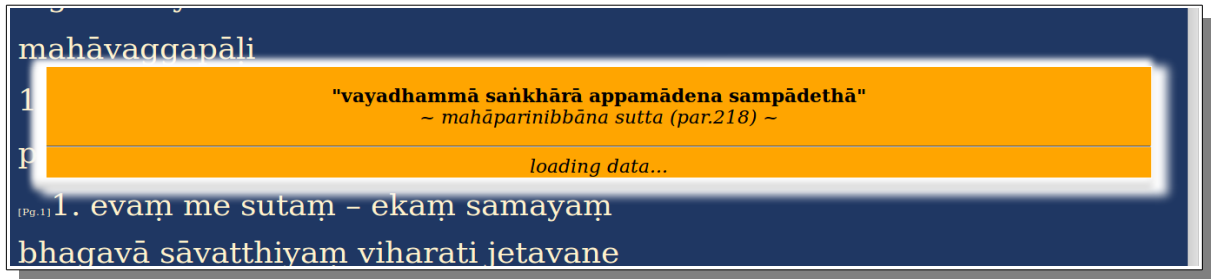
If even after doing so, your web-browser is still crashing when you open Pāli Tipiṭaka Projector, you may wish to clear your browser's cache, web browser's storage, and restart your device.

This “crashing” happens likely because of your device/web-browser doesn't have enough memory or powerful enough to load many Mb of dictionary data at once.

Content - Loading Data Message

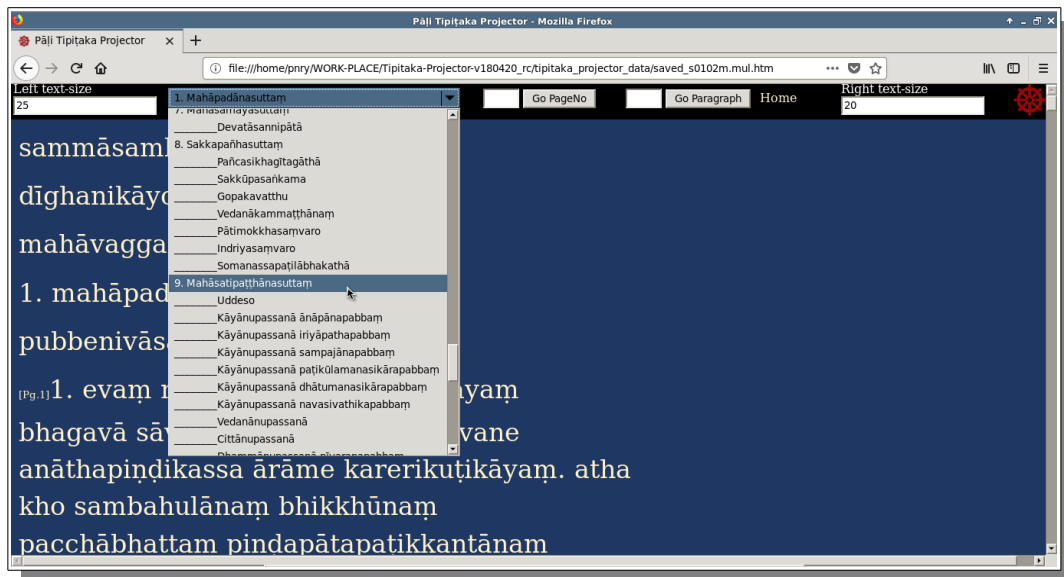
Whenever you see a loading message (**loading data...**), **please wait** until it disappears completely. Loading dictionaries and contents take time.


If you try a look up function before it finished loading all data, the app may not work correctly as expected.



For the **first** click event to look up, you may experience a slow-response from the app, but from the **second** click forwards, (but still depending on other factors), normally it should response faster than the first click.

Content – The Content Browser Menu



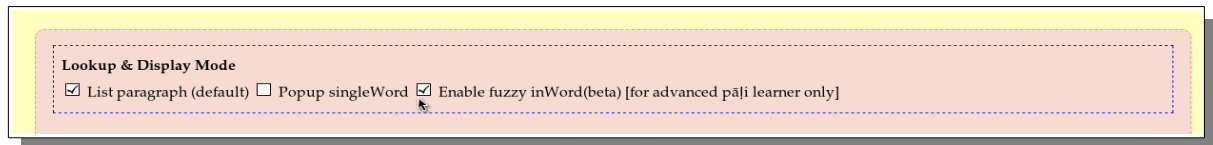
At any time, click on the symbol  at the **top-right corner** of the page, to open the content-browser menu, from there you can go to *chapter(vagga)/page/paragraph number* or adjust the *text-size, background color etc...*

The default text-size is declared in the file [projector_style.css](#). Default value for the left column text-size is 25pt, right column: 20pt; (quite large as this web-application was initially used with projector in a class).

Warning: Changing the text-size through this menu bar shall slow down the page load obviously each time you open a text, so if you know which value is suitable for your device screen, fix your text-size number in the file [projector_style.css](#) and use [02_RESET_Tipitaka_Projector.htm](#) to refresh the app (must do this reset step).

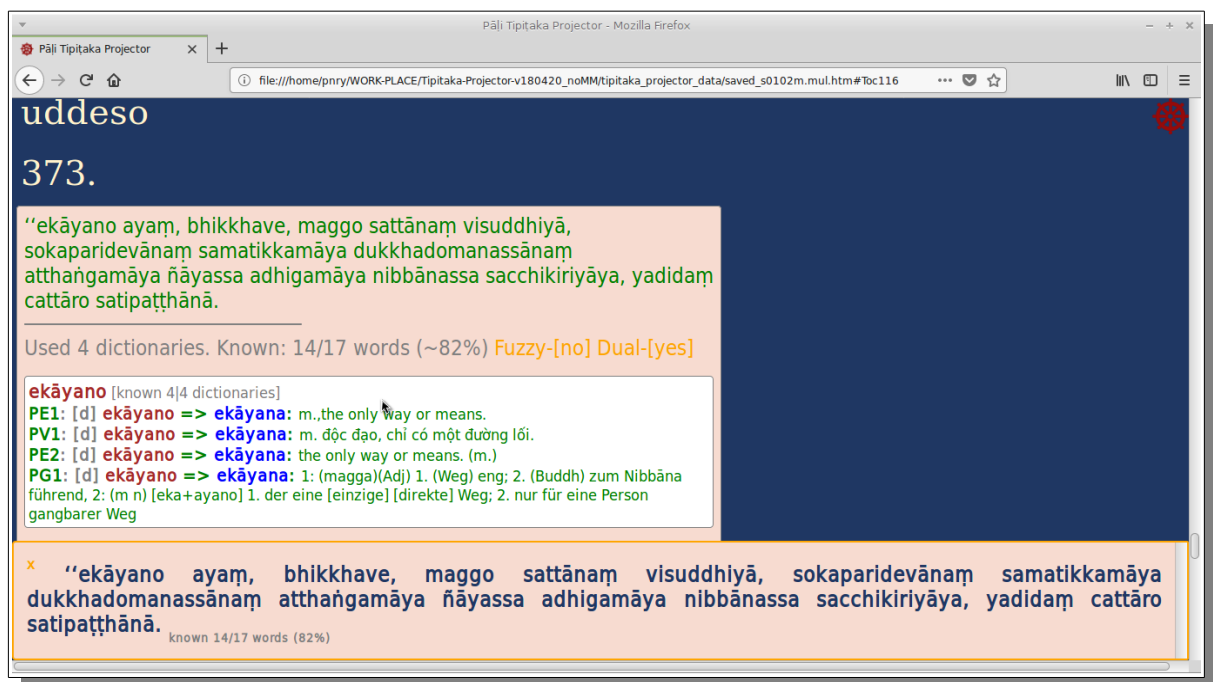
Dictionary - Understanding the List Paragraph Mode

Suppose that you are using the **List paragraph (default)** mode and enable **Enable fuzzy inWord(beta)** function.



Now open a content file and click on a paragraph/sentence. It shall show the meanings it can found/guess for all words of that paragraph/sentence.

Dictionary – Fuzzy-[yes] and Dual-[yes]



In this line:

Used 4 dictionaries. Known: 14/17 words (~82%) Fuzzy-[yes] Dual-[yes]

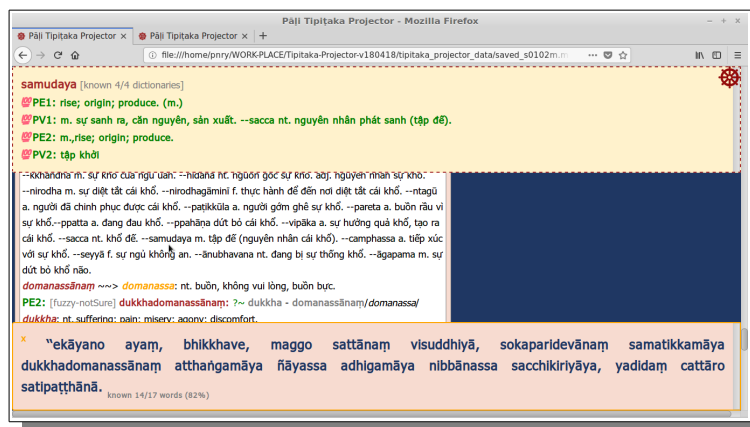
Fuzzy-[yes] and **Dual-[yes]** are “click-able”. At any time you can click on them and switch the setting to [yes] | [no] (= enable | disable).

Fuzzy-[yes]: informs that **Enable fuzzy inWord(beta)** is now enabled.

If you click on it, it shall switch to **Fuzzy-[no]** and our choice is saved. From next paragraph forwards, it shall stop the fuzzy [fuzzy-notSure] search function.

Dual-[yes]: you can click on any word in the definition area of a word entry to look up its meanings. If it is **Dual-[no]**, then it will remove the word entry containing that clicked word.

Dictionary – More On Dual-[yes]



Dual-[yes]: is useful when working with multi-language definitions. In this example, the word **—samudaya** in the definition area is clicked, another floating pop-up shall appear at the top of the page and showing its meanings.

This pop-up is **scrollable**.

So you can scroll it to view other definitions or to view a long definition.

Click on the pop-up itself to close it!

You can use this function to further look up a word that you don't understand (like foreign language in the definitions). Or any other words mentioned in the definition area itself.

Dictionary – 4 Types of look up Results

There are **four types** of look up results in this app. The better ones shall be displayed first in the definition area. They are displayed with this order: step/type of result: 1>2>3>4.

1. The one with **100** symbol means it could find that word in the dictionary.

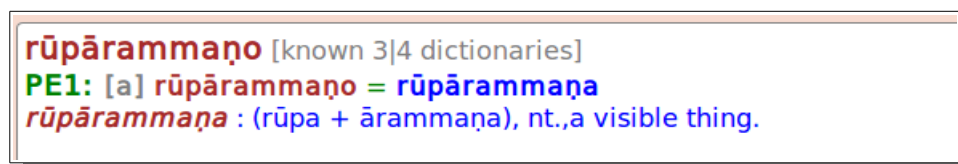
100 PE2: (nom. sing. of lma), m.; f.,this person.

[100 symbol here only means it found it in the dictionary, it does not cover the definition's quality :)]

2. The one with [d] means the dictionary does not have this word, but the app recognises it by using database and shows the meanings found in the dictionary.

PE1: [d] ekāyano => ekāyana: the only way or means. (m.)

3. The one with [a]: when the step no. 1&2 above are failed, it shall try to recognise the word by other ways.



4. The one with [fuzzy-notSure] means this is the last step when the step no. 1,2 3 above are all failed. It will try to split the word into smaller portions.

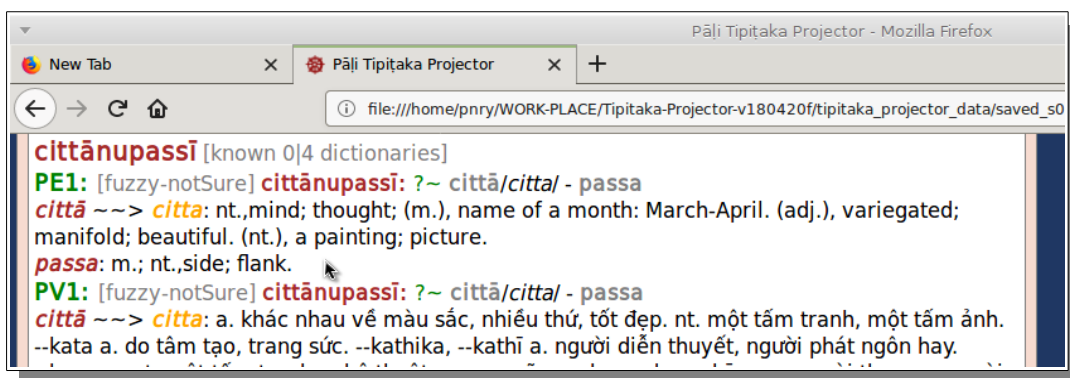
This function is not reliable. Sometimes it can give helpful information. Like this one:

```
sokaparidevāṇaṃ [known 0/4 dictionaries]
PE1: [fuzzy-notSure] sokaparidevāṇaṃ: ?~ soka - paridevāṇaṃ/parideva/
soka: grief; sorrow. (m.)
paridevāṇaṃ ~~~> parideva: wailing; lamentation. (m.)
```

But it does break a word wrongly sometimes! Please see the example below:

Dictionary – The Weakness of Fuzzy Search

1. Sometimes, it does break a word wrongly! Like this one:



The word **cittānupassī** should be recognised as **citta+ anupassī**.

For **PE1**, when the step no. 1,2, 3 above were all failed, [and the fuzzy in word function was enabled], it tried to break this word (but wrongly) into: **cittā +passa**.

It breaks wrongly in this case because it failed to recognise the **ā** in the middle that was formed by a+a in **citta+ anupassī**. If the word **cittānupassī** is written as **citta^anupassī**, then it could break in into **citta^a+anupassī** correctly since both **citta** and **anupassī** exist in the dictionaries.

This fuzzy search function is still a crude& simple approach, it needs more work to improve.

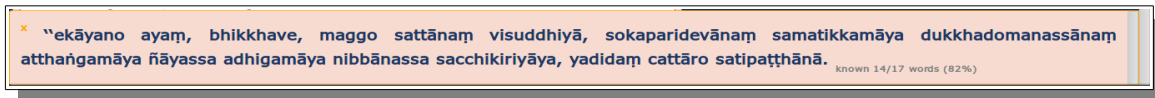
Please be reminded again that this function should be turned on only for **advanced pāli learners** when they have enough abilities to know whether this breaking is correct or not.


2. **Repeating of divided words.** You may see a divided word repeated 2 or more times. It might help to keep possible meanings, but in most cases, they will have the same meanings. Only in a few cases, they are different, for example, the word **mātikā** in this example:

```
dvemātikāpāḷi [known 0/4 dictionaries]
PE1: [fuzzy-notSure] dvemātikāpāḷi: ?~ dve/dvi/ - dve - mātikā/mātika/ - mātikā
dve ~~~> dvi: two. (the numeral) (adj.)
dve: the two. ((nom. plu. of dvi))
mātikā ~~~> mātika: connected with mother. (adj.)
mātikā: a water course; a table of contents; the code of Patimokkha. (f.)
```

Dictionary – The Floating Paragraph At The Bottom

- Words in the floating paragraph at the bottom are **linked** to the words in the definition area, click on a word here to scroll up the page to its definition.



- Click on the **x** (at the left corner, right before the text) to hide this floating paragraph.
-  symbol shall appear (at the bottom-left corner of the page), click on this image to show the floating paragraph again.
- This pop-up is also **scrollable** when the paragraph length is >50% height of the screen.

Dictionary – 3 Types of Removing Definition

There are **three types** of removing definition: removing definition of a **word entry**, **single paragraph/sentence**, and **all paragraphs/sentences**.

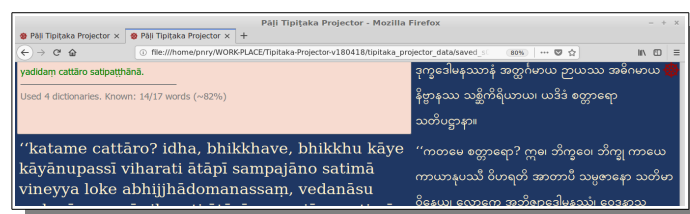
1- If **Dual** is **[no]**, click on any word inside a word entry, it will remove that **single word** entry containing the word that you have clicked on.

2- At the end of the definitions for each paragraph/sentence, click on the text **^^ Remove These Definitions ^^** to remove all definition entries of **this single paragraph** and scroll the page to [right after] **this** paragraph.

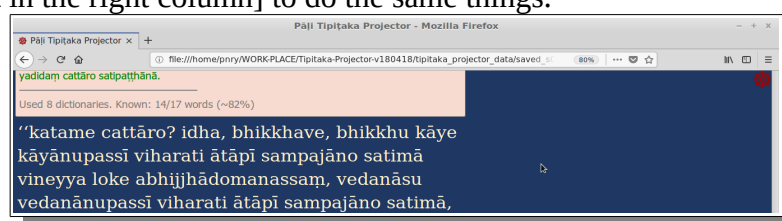


3 – At any time, click on the **right column**, it shall remove **all definitions** in the page and scroll up to the latest clicked paragraph (by time of click).

You can also use this function at any time to jump to your last clicked paragraph that you were working on.

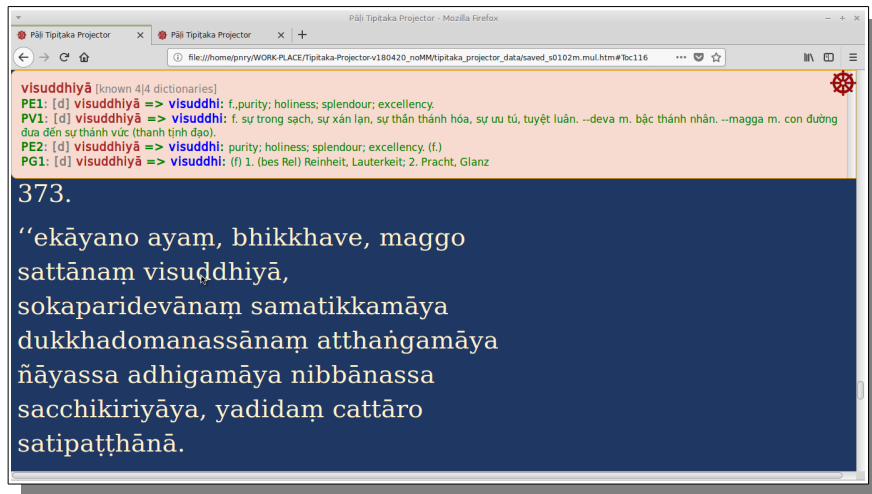


This function is also applicable when viewing with single Pāli Roman text, click on the right column [no text in the right column] to do the same things.



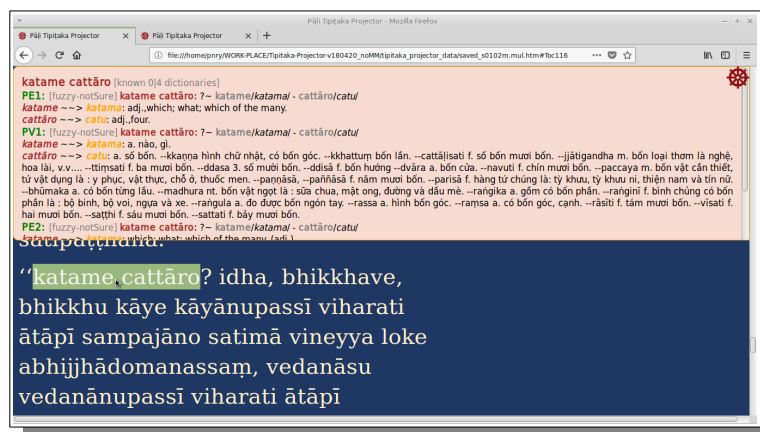
Dictionary – Pop-up singleWord Mode

For the **Pop-up singleWord mode**, it is more convenient for one who advanced in Pāli. It only shows meaning of single word.



- Click on a word in to look up its definitions. A floating pop-up shall appear at the top of the page.
- This pop-up is **scrollable**. So you can scroll it to view other definitions or to view a long definition.
- Click on any word inside that floating pop-up to look up its meanings. **It allows unlimited click.** It is similar to **Dual-[yes]** function, but it allows unlimited click and everything happens inside this floating pop-up.
- Click on the right column to close** this pop-up.

Tips: You can select a few characters of a single word or many words at once to look up them. On computer, it is more easier to do this with mouse/touch-pad?



It shall perceive all those words as a single word, and then break them into smaller portions to look up if necessary.

Software – For Developers

This software is very simple and a “non-professional” one. You may only need a short time to study its code and understand how it works. There are many supportive comments in the code files supporting your reading.

Basically, it uses *localStorage* of the web-browser to save user's settings and uses *Javascript/jQuery* to look up meanings in the dictionaries.

Tips 1: if web-browser does not support localStorage

If your web-browser/device does not support *localStorage*, but it supports *Javascript*, you may try this “work-around” to use the dictionary functions.

By default, it will not load any dictionary if the user did not check to enable it (except the core dictionary data). Now we change it to “load the dictionary” by default, by pass this setting.

For example with dictionary PE1. Open file **stlookup_jquery.js** with a text/code editor (**to avoid problems, please always open and save code files in this app with encoding in UTF-8**), then change the if condition from this:

```
var hpe1 = localStorage.getItem("hpe1");
if ((hpe1 == 1)) { dict_in_use++;
$.getScript("dictionary/pe1_Pali_English_sc2016_pi2en-maindata-v1.2.js");
}
```

To this: (adding a **!**) to the if condition:

```
var hpe1 = localStorage.getItem("hpe1");
if (!(hpe1 == 1)) { dict_in_use++;
$.getScript("dictionary/pe1_Pali_English_sc2016_pi2en-maindata-v1.2.js");
}
```

You can do the same things for other dictionaries. After that save **stlookup_jquery.js**.

Then open **02_RESET_Tipitaka_Projector.htm** to refresh the app (must do this “reset” step, otherwise this work-around may not work as expected).

Note: If there is error something like [in Opera/Chrome browser?]: “cross origin requests are only supported for protocol schemes: http, data, chrome, chrome-extension, https”...

Then the loading dictionary function cannot work. For chrome/chromium, you may try to run the browser with the option:

```
--allow-file-access-from-files
```

For example, with chromium on Linux, in the terminal run:

```
username@pclinux:~# chromium --allow-file-access-from-files
```

Then, open Pāli Tipiṭaka Projector again. If it is not OK, you may need to use another browser or include code to load the dictionaries directly into the content files.

Adding Dictionary Step 1 – Prepare a dictionary array in Javascript

You can add your dictionaries to Pāli Tipiṭaka Projector easily. One way to do this is to follow these 3 main steps:

To prepare a dictionary array in Javascript, one way to do this is using this simple format for your *Javascript* dictionary array:

(You may wish to study other dictionaries included in this software for references first.) For illustration purpose, we shall use the “dictionary name” as **pe123** [Pali-English, dictionary number 123rd]

```
var pe123 = {
  'one':['is 1'],
  'two':['is 2'],
  'it\s':['is it\s'],
  '...':['...'],
  'last':['this is the last entry']
};

//In both word and definition for each entry, any other ' must be
"escaped" by adding a \ before it, like this 'it\s':['is it\s']
//Please note that the last definition ending with ']' only, without a
comma ,
```

And save it with a name something like, for example: **pe123_newDictionary.js**

Step 2 – Add codes to the index files.

There are 8 htm files [index.htm, index2.htm, index3.htm, index4.htm, index5.htm, index6.htm, index7.htm, index8.htm]. Open an index file with a text/code editor:

2.1 Dictionary-In-Use Loader | Check to Enable

To add a check-box for your dictionary in the index file, find: Dictionary-In-Use Loader | Check to Enable

To quickly go to this area. Study other check-boxes under this line to understand how many variables needed to declare for your dictionary then add a check-box for your dictionary.

For example:

```
<input style="color:red;" id="rpe123" type="checkbox" name="hpe123"
onclick='enableDictionary_checkbox("hpe123")' /><label
onclick='enableDictionary("hpe123")' > PE123: [5 entries - 2 KB] More
Dictionary Name</label>
```

2.2 window.onload = function () {

Find: window.onload = function () {

And add one more autoReloadEnabledDict function for your dictionary. This will reload last choice of user.

For example: **autoReloadEnabledDict**("h**pe123**");

Repeat step 2.1 and 2.2 for all 8 index files.

Step 3 – Add codes to stlookup_jquery.js file

Open file **stlookup_jquery.js** with a text/code editor, then:

3.1 DICTIONARY VARIABLES AND PERFORM LOADING

Find: DICTIONARY VARIABLES AND PERFORM LOADING

Study other dictionaries' variables to understand what to do then add similar codes for your dictionary. For example:

```
var pe123 = ""; //The dictionary array must be always declared before
loading the script
var hpe123 = localStorage.getItem("hpe123"); //Check user setting:
enabled this dictionary or not
if ((hpe123 == 1)) { dict_in_use++;
$.getScript("pe123_newDictionary.js"); }
```

3.2 Call look up function for your dictionary here

Find: Call look up function for your dictionary here

Create and add codes for your dictionary. For example:

```
if (hpe123 == 1) { look upEachDict(pe123, voca_lowercase, "m_pe123", "u_pe123",
"PE123"); }
```

The last “**PE123**” in capital is the dictionary name that will be used when showing meanings. For example, it shall look like:

PE123: [d] **ekāyano** => **ekāyana**: the only way or means. (m.)

3.3 Optional: Font/Styles/Function for Your Specific Dictionary

This step is optional. Just in case your dictionary needs a special font to display it correctly or simply because you want to display it in another custom font/styles that you like.

You can also apply a specific function for your particular dictionary result (like converting it into another encoding, or calling a translating function to translate it into another language)

Suppose that you want to apply a custom font, then in the **step no 3.2** above, you need to add **one more parameter** to the code.

In some Myanmar dictionaries, they used Zawgyi font. So for illustration, we shall name this parameter as **ZawgyiFont**:

```
if (hpe123 == 1) { look upEachDict(pe123, voca_lowercase, "m_pe123", "u_pe123",
"PE123", "ZawgyiFont"); }
```

You need to do 2 more steps: to add a “class” to your dictionary results and declare this “class” styles in the file [projector_style.css](#)

Step 3.3.1 APPLY SPECIAL CUSTOM FONT FOR A SPECIFIC DICTIONARY

Find: APPLY SPECIAL CUSTOM FONT FOR A SPECIFIC DICTIONARY

Add a class to your dictionary result, for example we use a class named **ZawgyiFont**

(If you want to apply another function to your particular dictionary results, you can also do it in this step.)

```
if (specialfont == 'ZawgyiFont') {
  if (better_meaning.length > 0) { better_meaning = '<span class="ZawgyiFont">' +
better_meaning + '</span>'; }
  if (better_meaning_grammar.length > 0) { better_meaning_grammar = '<span
class="ZawgyiFont">' + better_meaning_grammar + '</span>'; }
  if (guess_meaning.length > 0) { guess_meaning = '<span class="ZawgyiFont">' +
guess_meaning + '</span>'; }
  specialfont = ""; //unset special custom font after use for a particular dictionary.
  Otherwise, all following definitions shall be affected
}
```

Step 3.3.2 Define your custom font and other styles for this class in [projector_style.css](#)

Now open file [projector_style.css](#) with a text/code editor.

Add more styles:

3.3.2.1 Define and add your font-family name and path to download the font. For example:

```
@font-face {
font-family: ZawgyiFont;
src: url('fonts/Myanmar/Zawgyi-One_V3.1.ttf');
}
```

Be noted that the src: url('fonts/Myanmar/Zawgyi-One_V3.1.ttf') is a relative path to the [projector_style.css](#) file, not the htm content file.

3.3.2.2 Finally define and add your class styles.

```
.ZawgyiFont {font-family:ZawgyiFont;background:orange;}
```

You can add many other styles to be applied for your dictionary, like *background*: etc...

Best wishes!

Software – Licenses & Credits

This app does include or use with/without modify source code and/or dictionary data that have been created by others.

For: Pāli Tipiṭaka Projector - Version 1.0 (180501)

No.	Items	Remarks
PROGRAMMING SOURCE CODES (Mainly Javascripts)		
1	jQuery JavaScript Library v3.3.1	jQuery JavaScript Library v3.3.1 https://jquery.com/
2	Script to Select Word On Click	Select Word on Click
3	Javascript functions: function toMyanmar(k),function toSinhala(l) function toDevar(l),function toThai(m)	Adapted from Adapted From Suttacentral Offline Website 2016 JS Data
4	Digital Tipiṭaka Text Website Adapted from a tipitaka digital text version from Vipassana Research Institute by Ven. Ariyavamsa (TW). (Used with permission from VRI for free distribution purpose.)	Adapted from: Tipitaka Digital Text Version From Vipassana Research Institute
FONTS		
1	DejaVuSansCondensed.ttf Version 2.37	Copyright (c) 2003 by Bitstream, Inc. All Rights Reserved. Copyright (c) 2006 by Tavmjong Bah. All Rights Reserved. DejaVu changes are in public domain See Check LICENSE in the folder fonts/DejaVu for more details
DICTIONARY DATA		
English Dictionary		
1	wn-Eng-Eng-Dictionary_WordNet_3.0_.js	Wordnet

Pali - English Dictionary		
1	000_wordbreakdata_Pali_Pali_grammar_Dictionary_extract_DPR_2018.js	000 Extracted and generated with Grammar Function from Digital Pali Reader (DPR) data by Ven. Ariyavamsa (TW)
2	pe1_Pali_English_sc2016_pi2en-maindata-v1.2.js Suttacentral Offline Website 2016 file: pi2en-maindata-v1.2.js	Adapted from Adapted From Suttacentral Offline Website 2016 JS Data
3	pe2_Pali_English_Dictionary_extract_DPR_2018.js	PE2 Extracted from Digital Pali Reader (DPR) data by Ven. Ariyavamsa (TW)
Pali - Vietnamese Dictionary		
1	* pv1 dictionary file: pv1_Pali_Viet_Dictionary_by_ngaiBuuChon_stardict.js "Pali Viet Dictionary Bản dịch của ngài Bửu Chơn."	PV1 Converted From Stardict File
2	* pv2 dictionary file: pv2_Pali_Viet_Abhi-Terms_by_ngaiTinhSu_Stardict.js	PV2 Converted From Stardict File
3	* pv3 dictionary file: pv3_Pali_Viet_Vinaya-Terms_by_VenGiacNguyenBhikkhu.js	PV3 Converted From Stardict File
Pali Indonesia Dictionary		
1	* pd1 dictionary file: pd1_Pali_Indo_sc2016_pi2id-maindata-v1.js Suttacentral Offline Website 2016 file: pi2id-maindata-v1.js	Adapted from Adapted From Suttacentral Offline Website 2016 JS Data
Pali - Germany Dictionary		
1	* pg1 dictionary file: pg1_Pali_Germany_sc2016_pi2de-maindata-v1.2.js Suttacentral Offline Website 2016 file: pi2de-maindata-v1.2.js	Adapted from Adapted From Suttacentral Offline Website 2016 JS Data
Pali - Chinese Dictionary		
1	* pc1 dictionary file: pc1_Pali_Zh_sc2016_pi2zh-maindata-v1.js	Adapted from Adapted From Suttacentral Offline Website 2016 JS Data

MORE DETAILS**Tipitaka Digital Text Version From Vipassana Research Institute**

Used with kind permission from VRI for free distribution purpose ONLY.

The tipitaka text used in this work is based on the digital tipitaka text version from Vipassana Research Institute.
Used with permission and gratefully acknowledgment.

This tipitaka digital text version copyright Vipassana Research Institute ("VRI"), Mumbai India.

Use by permission of VRI gratefully acknowledged.

jQuery JavaScript Library v3.3.1

You are free to use the Project in any other project (even commercial projects) as long as the copyright header is left intact.

```

/*!
 * jQuery JavaScript Library v3.3.1
 * https://jquery.com/
 *
 * Includes Sizzle.js
 * https://sizzlejs.com/
 *
 * Copyright JS Foundation and other contributors
 * Released under the MIT license
 * https://jquery.org/license
 *
 * Date: 2018-01-20T17:24Z
 */

```

Select Word on Click

Script Select Word on Click:

```

@ https://developer.mozilla.org/en-US/docs/Web/API/Selection
@ https://developer.mozilla.org/en-US/docs/Web/API/Selection/modify
@ https://developer.mozilla.org/en-US/docs/Web/API/Selection/isCollapsed
@ https://stackoverflow.com/questions/7563169/detect-which-word-has-been-
  clicked-on-within-a-text
@ http://jsfiddle.net/Vap7C/80/ <--this link was posted on
  https://stackoverflow.com/ by user stevendaniels
  (https://stackoverflow.com/users/515674/stevendaniels)
*/

```

Digital Pali Reader (DPR)

Used with kind permission from DPR

<https://sourceforge.net/projects/digitalpali/>
Related links <https://pali.sirimangalo.org/>

PV1 Converted From Stardict File

Source files: Form Stardict Dictionary Data
 StarDict's dict ifo file
 version=2.4.2
 wordcount=9965
 idxfilesize=174068
 bookname=Pāli-Viet (Bhikkhu Bửu Chơn)
 author=Bhikkhu Bửu Chơn
 description=convert to StarDict by KhoaBT btkhoa@yahoo.com
 date=2010.05.07
 sametypesequence=m

Related information:

<http://budsas.net/uni/u-tudien-palviet/tdpv-00.htm>

Tự điển Pāli-Việt giản lược

Hòa thượng Bửu Chơn
 Nāga Mahāthera
 PL. 2521 - DL. 1976

Dựa theo: "Concise Pāli-English Dictionary", A.P. Buddhadatta Mahāthera, Colombo (1968)

PV2 Converted From Stardict File

StarDict's dict ifo file
 version=2.4.2
 wordcount=3853
 idxfilesize=80413
 bookname=Abhidhamma - Bhikkhu Tịnh Sự
 author=Bhikkhu Tịnh Sự
 description=convert to StarDict by KhoaBT btkhoa@yahoo.com
 date=2010.05.07
 sametypesequence=m

Tu Dien Thuat Ngu Vi Dieu Phap
 ngài Tịnh Sự (phần ghi chú thuật ngữ)
 Ven. SANTAKICCA Mahā Thera (Đại Trưởng Lão TỊNH SỰ)

PV3 Converted From Stardict File

StarDict's dict ifo file
 bookname=Thuật ngữ tạng Luật Pāli-Viet (Bhikkhu Giác Nguyên)
 author=Bhikkhu Giác Nguyên
 description=convert to StarDict by KhoaBT btkhoa@yahoo.com
 date=2010.05.07
 sametypesequence=m

Related information

THUẬT NGỮ LUẬT TẠNG PĀLI
<http://budsas.net/uni/u-tudien-palviet/tlp-01.htm>
 Tỳ khưu Giác Nguyên biên dịch
 PL.2550 - DL. 2006
 */

Adapted From Suttacentral Offline Website 2016 JS Data

Used with kind permission from Suttacentral

Related information:

=====

File LICENSE.txt

SuttaCentral Copyright

=====

<<http://suttacentral.net/copyright>>

Material on SuttaCentral falls in three distinct categories in relation to copyright, which are detailed below. If you wish to assert copyright over any material on SuttaCentral, or wish to discuss copyright on SuttaCentral, please do not hesitate to contact us.

1. Original material created by SuttaCentral

All original material created by SuttaCentral is dedicated to the Public Domain as per Creative Commons Zero (CC0 1.0 Universal).

This includes all text, design, software, and images created by SuttaCentral or persons working for SuttaCentral, on the domain suttacentral.net, or any domains or subdomains owned or managed by SuttaCentral, unless otherwise specified.

You are invited to copy, alter, redistribute, present, perform, convey or make use of any or all of this material in any way you wish. We would appreciate a notice of attribution, but this is not necessary. Please let us know if you would like any assistance in making use of our materials.

2. Material created by others and made available for SuttaCentral

For certain of the material on SuttaCentral the copyright has been asserted by third parties. This includes most of the translated texts. In such cases the terms of the copyright are specified by the copyright asserter, who is usually the translator or original publisher. Such material is used by permission. The relevant copyright notices as specified by the asserter of copyright are included with the material.

3. Public domain material

The original texts of Buddhism in Pali, Chinese, Sanskrit, Tibetan, and other languages, are in the public domain. We believe that copyright assertions regarding such material have no legal basis. Nevertheless, we endeavor to use all materials with permission.

In addition, the reference data, including information on parallels, is not an "original creation" and as such does not fall within the scope of copyright.

Wordnet

Derived from Goldendict format, retrieved from:
<https://sourceforge.net/projects/goldendict/files/dictionaries>
https://sourceforge.net/projects/goldendict/files/dictionaries/WordNet3.0_1.0.zip

Wordnet License

```

1 This software and database is being provided to you, the LICENSEE, by
2 Princeton University under the following license. By obtaining, using
3 and/or copying this software and database, you agree that you have
4 read, understood, and will comply with these terms and conditions.:
5
6 Permission to use, copy, modify and distribute this software and
7 database and its documentation for any purpose and without fee or
8 royalty is hereby granted, provided that you agree to comply with
9 the following copyright notice and statements, including the disclaimer,
10 and that the same appear on ALL copies of the software, database and
11 documentation, including modifications that you make for internal
12 use or for distribution.
13
14 WordNet 3.0 Copyright 2006 by Princeton University. All rights reserved.
15
16 THIS SOFTWARE AND DATABASE IS PROVIDED "AS IS" AND PRINCETON
17 UNIVERSITY MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR
18 IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PRINCETON
19 UNIVERSITY MAKES NO REPRESENTATIONS OR WARRANTIES OF MERCHANT-
20 ABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE
21 OF THE LICENSED SOFTWARE, DATABASE OR DOCUMENTATION WILL NOT
22 INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR
23 OTHER RIGHTS.
24
25 The name of Princeton University or Princeton may not be used in
26 advertising or publicity pertaining to distribution of the software
27 and/or database. Title to copyright in this software, database and
28 any associated documentation shall at all times remain with
29 Princeton University and LICENSEE agrees to preserve same.

```

Software – New version & Feedback

<https://>