

* Gottfried <gottfried@posteo.de> [2022-12-14 19:40]:
> Hallo,
> ich wollte einen Puffer ausdrucken (das erste Mal)
> und es gab diese Fehlermeldung:

So far best tool to print Emacs buffers that I have found is the command `paps' as it supports correctly Unicode, also emoticons. Thus you may try by installing `paps' tool from your Operating System distribution.

Das beste Werkzeug zum Drucken von EMACS-Puffern, die ich gefunden habe, ist der Befehl "paps", weil es richtig Unicode unterstützt, auch Emoticons. Daher können Sie es versuchen, indem Sie das "PAPS"-Tool aus Ihrer Betriebssystemverteilung installieren.

dov/paps: A text to postscript converter through pango:
<https://github.com/dov/paps>

Ich verwende folgende Funktionen:

```
(defun rcd-command-output-from-input (program input &rest args)
  "Return output string from PROGRAM with given INPUT string and optional ARGS."
  (let* ((output (with-temp-buffer
                   (insert input)
                   (apply #'call-process-region nil nil program t '(t nil) nil args)
                   (buffer-string))))
    output))
```

I am too often using landscape, you can adjust it for your needs.

```
(defun rcd-paps-process (text &optional format title &rest args)
  (let* ((format (or format "pdf"))
         (title (or title (buffer-name)))
         (length (length title))
         (max 45)
         (title (if (> length max) (substring title 0 max) title)))
    (apply 'rcd-command-output-from-input "paps" text "--format" format "--landscape" "--title" title "--header" args)))

(defun string-to-file-force (string file)
  "Prints string into file, matters not if file exists. Return FILE as file name."
  (with-temp-file file
    (insert string))
  file)

(defun rcd-paps-text-to-pdf (text &optional file-name title)
  (let* ((output (rcd-paps-process text "pdf" title))
         (file-name (or file-name (concat (file-name-as-directory (getenv "HOME")) (read-string "File name without .pdf extension: ") ".pdf"))))
    (string-to-file-force output file-name)))

(defun rcd-paps-buffer-to-pdf-view ()
  (interactive)
  (let ((output (rcd-paps-process-buffer))
        (file-name (rcd-temp-file-name nil "pdf")))
    (sleep-for 1)
    (start-process "evince" nil "evince" (string-to-file-force output file-name))))
```



Jean

Take action in Free Software Foundation campaigns:

<https://www.fsf.org/campaigns>

In support of Richard M. Stallman

<https://stallmansupport.org/>