The limecv document class*

Olivier Pieters
me(at)olivierpieters(dot)be

v0.1.3 from 2017/09/12

Contents

1 Introduction **1**

2 Requirements **2**

3 General Macros and Document Class Options **2**
3.1 Internationalisation 2

4 Sidebar **3**

5 Main Content **6**

6 Cover Letter **9**

7 Change Style and Layout **10**

8 Example **13**

9 Implementation **13**

1 Introduction

This document class is designed to facilitate easy development of curriculum vitæ (CV). A set of predefined macros and environments are available for most common sections on a CV. These special elements have been designed to ease quick creation. This document class was co-designed with a business card, which can be found on GitHub: `https://github.com/opieters/business-card`.

The design of this CV is split up in three parts, illustrated by fig. 1. Each of these parts that make up this CV template will be detailed in the sections below.

*This document corresponds to limecv v0.1.3, dated 2017/09/12.

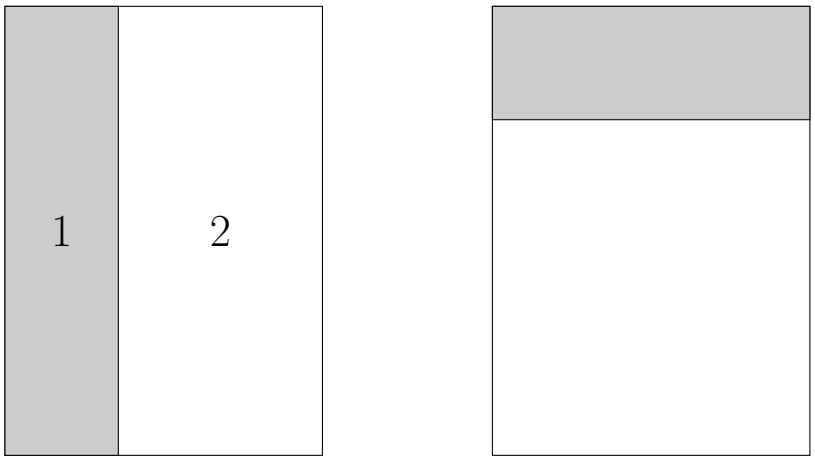

Figure 1: Illustation of a basic template. The image to the left depicts the actual CV: sidebar to the left (1) with main content on the right (2). The image to the right depicts the cover letter design.

2 Requirements

It is advised to use the **XeLaTeX** engine. However, there is also support for **LaTeX** and **LuaLaTeX**. In these cases, fonts might be different than intended and side bearings will not be corrected, since only the XeLaTeX engine has access to these. In the subsequent sections, it will always be assumed that the XeLaTeX compiler is used (unless noted otherwise).

Any font can be used, though by default the **Fira** [1] font is used. This should be installed and accessible by the typesetting system. If another font is desired, it can be overwritten using the `sansfont` document class options and `\cvMainFont` command. The mono spaced font can be changed by means of the `monofont` class option.

FontAwesome[2] is the icon font used. This font should also be available and cannot be replaced by another icon font. Most modern TeX distributions ship with this font since they provide the `fontawesome` package.

3 General Macros and Document Class Options

3.1 Internationalisation

`\cvSetLanguage` Currently, there is only internationalisation support for English and Dutch. The default language is English and can be modified using the `\cvSetLanguage` macro. Possible arguments are: `english` and `dutch`. If you want to use a different language, you need to modify the keys that hold the different section titles. These

[1] `https://github.com/mozilla/Fira`
[2] `http://fontawesome.io`

can be in the source code. The following snippet illustrates how this can be done for French:

```
\pgfkeys{/@cv/names/education = Formations}
\pgfkeys{/@cv/names/skills = Comp tences}
```

4 Sidebar

The sidebar should contain personal information such as your name, job title (or industry or similar), contact information, small bio, interests and language skills. Special environments and commands have been defined for each of these sections and will be described below.

Everything that should be inside the sidebar should be placed in the `cvSidebar` environment. This environment is placed on the left side of the page by default. If it should be typeset on the right side, use the starred version (`cvSidebar*`)

The following environments are available inside the sidebar environment: `cvProfile`, `cvContact`, `cvLanguages`, `cvInterests` and `cvProjects`.

`\cvID` This command typesets a picture (in a circle) with name and position underneath it. The argument order is: `\cvID{`⟨*first name*⟩`}{`⟨*last name*⟩`}{` ⟨*picture location*⟩`}{`⟨*job position*⟩`}`. Empty fields are allowed for the third and fourth arguments. No picture and no job position will then be typeset. Example code:

```
\cvID{John}{Doe}{profile_picture}{Broker}
```

`cvProfile` This environment contains a brief profile description or biography. No additional arguments are allowed. Example code:

```
\begin{cvProfile}
  A short biography goes here.
\end{cvProfile}
```

`cvContact` All the contact information goes here. Inside this environment, the following commands are available:

`\cvContactAddress`
- `\cvContactAddress{`⟨*address*⟩`}` typesets an address. How this address should be typeset exactly, is left to the user. The use of line breaks (`\\`) is allowed;

`\cvContactEmail`
- `\cvContactEmail{`⟨*link*⟩`}{`⟨*email address*⟩`}` typesets an email address. The link variable should be a something like `mailto:john@doe.tld`. Clicking on the email address will then automatically open the default email client with this address as recipient. If the link argument is left empty, no link will be created.

`\cvContactPhone`
- `\cvContactPhone{`⟨*mobile phone number*⟩`}` typesets a mobile phone number.

`\cvContactWebsite`
- `\cvContactWebsite{`⟨*link*⟩`}{`⟨*website URL*⟩`}` typesets a website. The link variable should be a something like `https://johndoe.tld`. Clicking on the

website will then automatically open the default web browser. If the link argument is left empty, no action will be performed upon clicking on the website.

`\cvContactGithub`

- `\cvContactGithub{`⟨*link*⟩`}{`⟨*username*⟩`}` typesets a GitHub profile. The link variable should be a valid link to the GitHub profile (for example `https://github.com/johndoe`). Clicking on the username will then automatically open the default web browser. If the link argument is left empty, no action will be performed upon clicking on the website.

`\cvContactLinkedin`

- `\cvContactLinkedin{`⟨*link*⟩`}{`⟨*username*⟩`}` typesets a LinkedIn profile. The link variable should be a link to your LinkedIn profile homepage (for example `https://www.linkedin.com/in/johndoe/`). Clicking on the username will then automatically open the default web browser. If the link argument is left empty, no action will be performed upon clicking on the website.

`\cvContactTwitter`

- `\cvContactTwitter{`⟨*link*⟩`}{`⟨*username*⟩`}` typesets a Twitter profile. The link variable should direct to your Twitter profile. An example link looks as follows: `https://twitter.com/johndoe`. Clicking on the username will then automatically open the default web browser. If the link argument is left empty, no action will be performed upon clicking on the website.this address as recipient. If the link argument is left empty, no link will be created.

`\cvContactKeybase`

- `\cvContactKeybase{`⟨*link*⟩`}{`⟨*fingerprint*⟩`}` typesets a Keybase fingerprint (and account). The link variable should be a link to the KeyBase profile (e.g. `https://keybase.io/johndoe`). Clicking on the fingerprint will then automatically open the default web browser. If the link argument is left empty, no action will be performed upon clicking on the website.

A full example:

```
\begin{cvContact}
  \cvContactAddress{Some Street 78\\B-9000 Ghent\\
     Belgium}
  \cvContactEmail{mailto:john@doe.tld}{john@doe.tld}
  \cvContactPhone{+1 781 555 1212}
  \cvContactWebsite{https://doe.tld}{doe.tld}
  \cvContactGithub{https://github.com/johndoe}{johndoe}
  \cvContactLinkedin{https://www.linkedin.com/in/johndoe
     /}{johndoe}
  \cvContactTwitter{https://twitter.com/johndoe}{
     @johndoe}
  \cvContactKeybase{https://keybase.io/johndoe}{\texttt{
     AAAA 5555 BBBB FFFF}}
\end{cvContact}
```

If you wish to add contact information that is not available by default, you can extend the command using two internal commands: `\cv@ContactTemplateLink` and `\cv@ContactTemplate`. See the source code for usage instructions.

cvLanguages
\cvLanguage

This environment is used to showcase language skills. The `\cvLanguage{`⟨*language*⟩`}{`⟨*skill level*⟩`}` should be used inside this environment. The skill level is a real value with a maximum value of 5. If higher values are used, the result will not be typeset properly. An example is included below.

```
\begin{cvLanguages}
  \cvLanguage{English (native)}{5}
  \cvLanguage{German (B1)}{3}
  \cvLanguage{Spanish}{3}
\end{cvLanguages}
```

cvInterests
\cvInterestsPersonal
\cvInterestsProfessional
\cvInterest

Typeset interests (can be both professional and personal) using `cvInterests`. By default it just typesets a list of items in the long format (`long`). The short format can be activated by passing the `short` option to the environment. Inside this environment, three commands can be used: `\cvInterestsPersonal`, `\cvInterestsProfessional` and `\cvInterest`. `\cvInterestsPersonal` and `\cvInterestsProfessional` add optional sections inside this environment to differentiate between personal and professional interests respectively. Both macros have no options nor arguments. The `\cvInterest{`⟨*icon*⟩`}{`⟨*interest*⟩`}` command takes an icon and interest as arguments.

Examples that illustrate the different options are depicted below:

```
\begin{cvInterests}
  \cvInterestsPersonal
  \cvInterest{\faTrain}{model trains}
  \cvInterest{\faFlask}{(applied) sciences}
  \cvInterest{\faSuitcase}{travelling}
  \cvInterestsProfessional
  \cvInterest{\faGraduationCap}{machine learning}
  \cvInterest{\faCogs}{robotics}
\end{cvInterests}
```

```
\begin{cvInterests}[short]
  \cvInterestsPersonal
  \cvInterest{\faTrain}{model trains}
  \cvInterest{\faFlask}{(applied) sciences}
  \cvInterest{\faSuitcase}{travelling}
  \cvInterest{\faCamera}{photography}
  \cvInterest{\faGamepad}{gaming}
  \cvInterest{\faMusic}{music}
\end{cvInterests}
```

cvProjects
\cvProject

If you have interesting (side) projects that are relevant for your CV, you can list them using the `cvProjects` environment. Inside this environment you can use the `\cvProject[`⟨*options*⟩`]{`⟨*name*⟩`}{`⟨*description*⟩`}` macro to list all your projects. The only options currently allowed in ⟨*options*⟩ are an image (using `image`) and a URL (using `link`). This image must be an external file and the user must handle its size through `width` or `height`. Example usage:

```
\begin{cvProjects}
  \cvProject[image=clock,width=1cm]{yanic}{An IoT nixie
     clock.}
  \cvProject{\texttt{limecv}}{A \LaTeX\ document class
     for curriculum vit\ae.}
\end{cvProjects}
```

It is currently not possible to extend the sidebar with additional environments. To add your own, look at the source code and create your own LaTeX-style hack.

5 Main Content

The main content section includes details on your education, experience, skills, references and more. Several environments have been designed to suit specific needs. These will be discussed next.

`cvMainContent` Everything in the main content section should be encapsulated in the `cvMainContent` environment. This environment defines four new environments: `cvEducation`, `cvExperience`, `cvSkills` and `cvReferences`. These four environments are self explanatory in terms of functionality. We will detail each of these environments.

Note that `cvMainContent` also has a starred variant (`cvMainContent*`). The function is similar to `cvSidebar*`, in the sense that it places everything to the left instead of the default right location.

`cvEducation` The education environment creates a timeline styled list of your education. Individual education items should be listed by means of the `\cvItem{`⟨*details*⟩`}` macro which is available within this environment. Instead of forcing a specific layout structure, it was preferred to leave the actual mark-up to the end user. *All* information concerning a single education should be passed to this single argument. However, the user is always welcome to create his own styling macro that takes multiple arguments. This is illustrated by the examples below.

Listing 1: `cvEducation` *without* special user mark-up command.

```
\begin{cvEducation}
  \cvItem{Evening class: Chinese\\
  Some School, City. September 2015 -- June 2016\\
  Achieved A2 language skill in Chinese (Mandarin).}
  \cvItem{Bachelor of Science in Biochemistry and
     Biotechnology\\
  University, City. September 2009 -- June 2012\\
  General training in the basic sciences and the
     molecular life science.}
  \cvItem{Master of Science in Biochemistry and
     Biotechnology\\
  University, City. September 2012 -- June 2015\\
  Acquisition of insight into and knowledge of
     possibilities for application in the area
```

```
  of biochemistry and biotechnology, specific with
     applications in biomedical application
  and due problem-solving reasoning skills.}
\end{cvEducation}
```

Listing 2: `cvEducation` *with* special user mark-up command.

```
% in preamble:
\newfontfamily\firaMedium{Fira Sans Medium}
\NewDocumentCommand{\cvEducation}{mmm}{{\firaMedium
    #1}\\ #2\\ \emph{#3}}
% in document:
\begin{cvEducation}
  \cvItem{\cvEducation{Evening class: Chinese}%
            {Some School, City. September 2015 -- June
               2016}%
            {Achieved A2 language skill in Chinese (
               Mandarin).}}
  \cvItem{\cvEducation{Bachelor of Science in
     Biochemistry and Biotechnology}%
            {University, City. September 2009 -- June
               2012}%
            {General training in the basic sciences and
               the molecular life science.}}
  \cvItem{\cvEducation{Master of Science in Biochemistry
      and Biotechnology}%
            {University, City. September 2012 -- June
               2015}%
            {Acquisition of insight into and knowledge
               of possibilities for application
             in the area of biochemistry and
                biotechnology, specific with
                applications
             in biomedical application and due problem-
                solving reasoning skills.}}
\end{cvEducation}
```

`cvExperience` `cvExperience` works very similar to `cvEducation`. If follows the exact same structure and has the same design philosophy where you should use `\cvItem` inside this environment to typeset the individual items in a timeline style. Listing 3 illustrates this with an example.

Listing 3: `cvExperience` code example.

```
\begin{cvExperience}
  \cvItem{Student Job\\
    \textsc{\selectfont Company X}, Location X. Summer
       2010\\
```

```
    Integer tincidunt dapibus consectetur. Nullam
       tristique aliquam luctus. Sed ut ante velit.
    Nulla pharetra maximus lacus at elementum.
       Suspendisse sodales consectetur metus, sit amet
    ultricies ipsum ultrices ut.};
  \cvItem{Internship\\
    \textsc{Company Y}, Location Y. June 2012 -- August
       2012\\
     Lorem ipsum dolor sit amet, consectetur adipiscing
        elit. Morbi dictum cursus sapien, id
     eleifend mi pellentesque id. Etiam lobortis eu odio
         a sodales. Phasellus ut dolor feugiat,
     lacinia lectus in, blandit metus. Fusce lacinia
        dolor et metus gravida pulvinar sit amet
     et ex.};
  \cvItem{Internship\\
    \textsc{Company Z}, Location Z. August 2014 --
       September 2014\\
    Lorem ipsum dolor sit amet, consectetur adipiscing
       elit. Morbi dictum cursus sapien, id
    eleifend mi pellentesque id. Etiam lobortis eu odio
       a sodales. Phasellus ut dolor feugiat,
    lacinia lectus in, blandit metus.  Fusce lacinia
       dolor et metus gravida pulvinar sit amet
    et ex. Suspendisse vestibulum, leo malesuada
       molestie maximus, sem risus ornare elit, vitae
    sodales felis elit in ipsum.};
\end{cvExperience}
```

`cvSkills` The skills section is contained within the `cvSkills` environment. This environment typesets your skills on a 5-level (discrete) scale. These are divided into two columns. To that end, two macros are available: `cvSkillOne` and `cvSkillTwo`.

`cvSkillTwo` `cvSkillTwo{`⟨*skill level*⟩`}{`⟨*skill*⟩`}{`⟨*skill level*⟩`}{`⟨*skill*⟩`}` typesets a row of two

`cvSkillOne` skills. If you have an odd number of items, `cvSkillOne{`⟨*skill level*⟩`}{`⟨*skill*⟩`}` should be used. An example of a skill-list can be found in listing 4.

Listing 4: Illustration of the `cvSkills` environment.

```
\begin{cvSkills}
  \cvSkillTwo{5}{MATLAB}{5}{\LaTeX}
  \cvSkillTwo{4}{Python}{4}{VHDL}
  \cvSkillTwo{4}{Microsoft Office}{4}{macOS}
  \cvSkillOne{3}{C, C++}
\end{cvSkills}
```

`cvReferences` The final section is intended to list all your references. These go inside the `cvReferences` environment. The enumeration of the different items

`\cvAddReference` should be done using the `\cvAddReference{`⟨*information*⟩`}` macro. The follow-

ing keys are available: `name`, `company`, `job`, `address line 1`, `address line 2`, `address line 3`, `mobile phone`, `work phone` and `email`. These are all optional arguments and will be typeset consistently between the two references per row. When adding two references, the left one should be specified first using `\cvAddReference*`. This will store the values and typeset them when adding the second reference through `\cvAddReference`. Note that `\cvAddReference` should be used if only one reference needs to be typese. Listing 5 illustrates the usage of this environment.

Important remark: the comment after the usage of `\cvAddReference` is required! Otherwise, spacing will not be as intended.

Listing 5: `cvReferences` code example.

```
\begin{cvReferences}
  \cvAddReference{%
    name=Jane Smith,
    company=Company ABC Co.\ Ltd.,
    job=Job title,
    address line 1=Street lane 2,
    address line 2=B-1150 Brussels,
    mobile phone=+1 781 555 1212}% <<-- Important!!!
\end{cvReferences}
```

6 Cover Letter

A final (optional) part of a CV is the cover letter. This is a fairly simple part to create design wise, but probably the hardest to write in an actual CV.

`cvCoverLetter` The cover letter environment is `cvCoverLetter` and contains all the cover letter details. It will automatically add a header with your name and position based on the information filled in in `\cvID`.

`\cvBeneficiary` The `\cvBeneficiary{`⟨*options*⟩`}` macro offers a convenience wrapper to typeset the beneficiary. Possible options are `name`, `position`, `company`, `address line 1`, `address line 2` and `address line 3`. The remainder of the cover letter design is up to the user. An example design can be found in listing 6.

`\cvFullName` `\cvFullName` typeset the authors name based on the data provided in `cvID`.

Listing 6: `cvCoverLetter` code example.

```
\section{Cover Letter}

\begin{cvCoverLetter}

\cvBeneficiary{%
  name=Jane Smith,
  position=Position,
  company=Company,
  address line 1=Address line 1,
```

```
  address line 2=Address line 2}

Dear Miss.\ Smith

\vspace{\baselineskip}
\lipsum[1-3]
\vspace{\margin}

\cvFullName

\end{cvCoverLetter}
```

7 Change Style and Layout

To change the style, one can redefine the colours, lengths and TikZ styles used throughout the document class.

By default the following colours are defined:

- `cvGreen` default dark green colour (hex value: `357F2D`)
- `cvGreenLight` light green colour, for larger shapes and backgrounds (hex value: `B8E4B3`)
- `cvDark` default text colour (hex value: `2F3142`)
- `cvRed` currently not used (hex value: `7F2D35`)
- `cvAccent` lighter grey shade for accented text (hex value: `474A65`)

To change these, use the code from listing 7 in the preamble:

Listing 7: Colour editing.

```
\definecolor{cvDark}{HTML}{000000}
```

By overwriting the following styles, one can change the appearance of multiple TikZ nodes. The following styles are available.

- `interesticon`
- `interesttext`
- `eventdottext`
- `invisibletimedot`
- `timedot`
- `eventdot`
- `contactIcon`

- `contactText`
- `headerIcon`
- `skillLevel`
- `languageText`
- `skillText`
- `progressArea`
- `progressBar`
- `sectionTitle`
- `sectionEduText`

Adding your own styles is easy with the code from listing 8.

Listing 8: Change TikZ style.

```
\tikzset{
  interesticon/.style = {},
}
```

Spacing used inside the document can also be modified by the user. The following lengths have been defined:

- `\cvColSep` Distance between columns of `cvSkills`. Default value: `1cm`.
- `\cvNodeSep` Not used currently. Default value: `0.75cm`.
- `\cvTimeDotDiameter` Size of the dots in a timeline. Default value: `3pt`.
- `\cvMargin` Margin around the entire page. Default value: `1cm`.
- `\cvSideWidth` Width of the sidebar. Default value: `0.3\paperwidth-\cvMargin`.
- `\cvMainWidth` Width of the main content section. Default value: `\paperwidth-4\cvMargin-\cvSide`
- `\cvTimeDotSep` Distance between timeline dot and text. Default value: `0.4cm`.
- `\cvStartEndSep` Not used. Default value: `0.3cm`.
- `\cvItemSep` Distance between `cvItem` entries. Default value: `0.4cm`.
- `\cvTableSepWidth` Distance between two columns of `cvReferences` environment. Default value: `0.3cm`.
- `\cvCoverLetterHeight` Height of the coloured bar at the top of the cover letter. Default value: `\cvSideWidth`.

- `\cvCoverLetterWidth` Width of the cover letter text. This can be used to improve to layout. Default value: `\cvMainWidth+3\cvMargin`.
- `\cvPictureWidth` Diameter of the profile picture. Default value: `4cm`.
- `\cvProgressAreaWidth` Length of a progress bar for the languages section. Default value: `5em`.
- `\cvProgressAreaHeight` Height of a progress bar for the languages section. Default value: `0.75em`.
- `\cvSectionSep` Distance between sections of the main content section. Default value: `0.6cm`.
- `\cvSectionSBSep` Distance between sections of the sidebar section. Default value: `4pt`.
- `\cvTitleLineWidth` Line width of horizontal line below section title. Default value: `2pt`.
- `\cvTitleLineSpacing` Distance from bottom of the titles of the main content section to the horizontal centre of line. Default value: `0.1cm`.
- `\cv@projectSep` Internal length. Automatically set to `\cvItemSep`.
- `\cvProjectDetailsSep` Internal spacing between details of a single project. Default value: `0.2cm`.
- `\cvInterestDetailsSep` Distance between items in case the `long` options is used for the `cvInterests` environment. Default value: `4pt`.
- `\cvContactItemSep` Distance between items in the `cvContact` environment. Default value: `5pt`.
- `\cvCoverLetterLineWidth` Line width of line below cover letter name. Default value: `3pt`.
- `\cvCoverLetterPositionSpacing` Distance between name and position in the cover letter. Default value: `10pt`.
- `\cvSBSectionLineWidth` Line width of left and right lines next to the titles of in the sidebar. Default value: `1pt`.
- `\cvCoverLetterLineSpacing` Distance between name and line below it in the cover letter. Default value: `0pt`.
- `\cvIDNameSep` Distance between first and last name. Default value: `0.1cm`.
- `\cvPositionSep` Distance before and after the position. Default value: `0.5cm`.
- `\cvSkillSep` Distance between skills. Default value: `1ex`.

- `\cvHeaderIconWidth` Automatically set, currently has no visible effect on document. Default value: `0pt`.

The fonts can also be changed, either by using the `sansfont` and `monofont` class options, or by using the default ways provided by LaTeX, XeLaTeX and LuaLaTeX. Note that the `fontspec` package is already loaded by the class, so you can directly use `\setmainfont` in the preamble if you are using the XeLaTeX or LuaLaTeX engine.

8 Example

The source code of a typical CV document can be found in the examples folder. Figures 2 and 3 depict the resulting PDF documents.

9 Implementation

limecv.cls

Copyright 2017 Olivier Pieters

A (Xe/Lua)LaTeX document class

This work may be distributed and/or modified under the conditions of the LaTeX Project Public License, either version 1.3 of this license or (at your option) any later version. The latest version of this license is in http://www.latex-project.org/lppl.txt and version 1.3 or later is part of all distributions of LaTeX version 2005/12/01 or later.

This work has the LPPL maintenance status 'maintained'.

The Current Maintainer of this work is Olivier Pieters. Contact the maintainer at: me (at) olivierpieters (dot) be.

Include minimal required packages
process key-value class options

```
\RequirePackage{kvoptions}
```

Process options of this class.

```
\DeclareBoolOption[false]{print}
\newif\ifcv@fontPath
\cv@fontPathfalse
\DeclareStringOption{path}
\AddToKeyvalOption*{path}{\cv@fontPathtrue}
\DeclareStringOption[Fira Sans]{sansfont}
\DeclareStringOption[Fira Mono]{monofont}
\ProcessKeyvalOptions*
```

Process general class options
pass document options to article class, which forms base of this document class

Figure 2: Example CV (scaled).

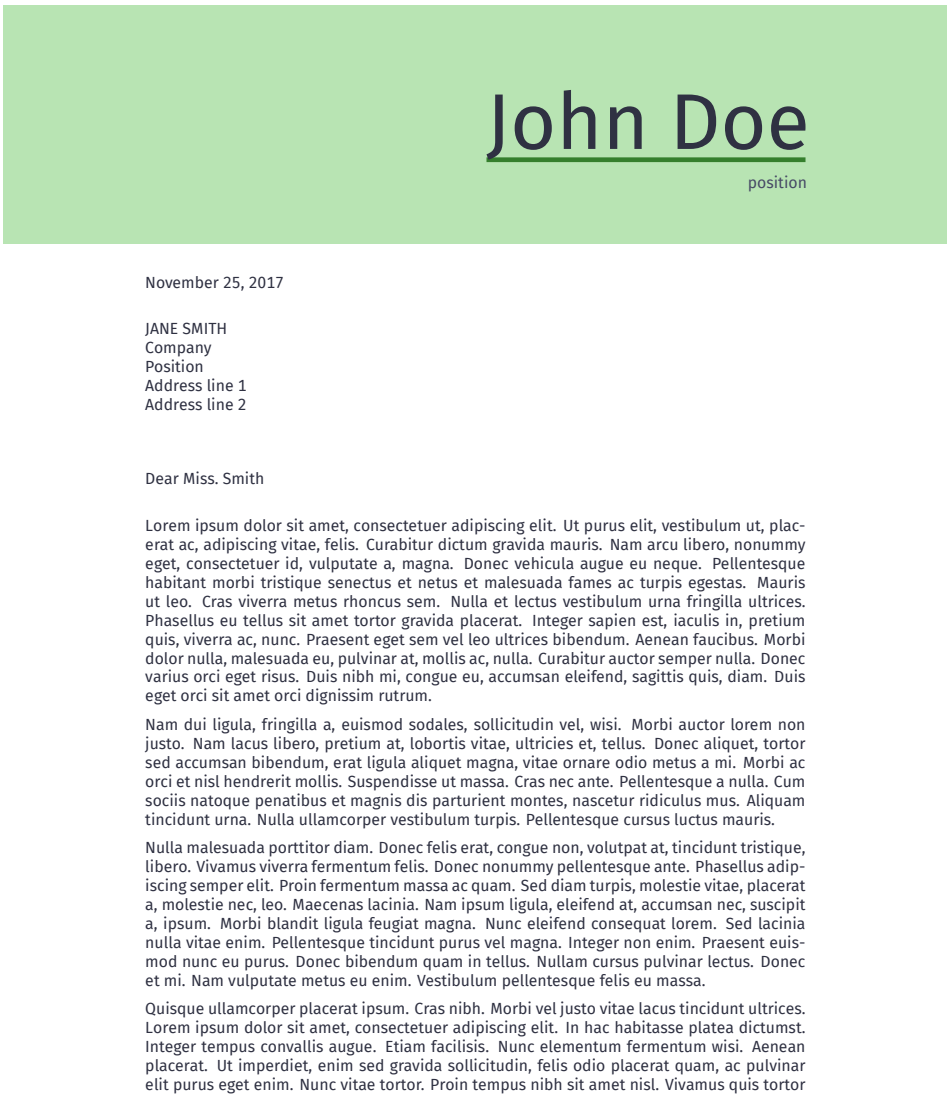

Figure 3: Example cover letter (scaled). Figure not included.

```
\DeclareOption*{%
  \PassOptionsToClass{\CurrentOption}{article}%
}
\ProcessOptions\relax
\LoadClass{article}
```

Include required packages
check if XeTeX is used

```
\RequirePackage{ifxetex}
```

check if LuaTeX is used

```
\RequirePackage{ifluatex}
```

Advanced command definitions

```
\RequirePackage{xparse}
```

complex length calculations

```
\RequirePackage{calc}
```

colour options

```
\RequirePackage{xcolor}
```

advanced table options

```
\RequirePackage{tabularx}
```

links

```
\RequirePackage{hyperref}
\RequirePackage{url}
```

remove paragraph indent

```
\RequirePackage{parskip}
```

string operations

```
\RequirePackage{xstring}
```

key-value parsing within document

```
\RequirePackage{xkeyval}
```

Define colours

```
\definecolor{cvGreen}{HTML}{357F2D}
\definecolor{cvGreenLight}{HTML}{b8e4b3}
\definecolor{cvRed}{HTML}{7f2d35}
\definecolor{cvDark}{HTML}{2F3142}
\definecolor{cvAccent}{HTML}{474A65}
```

Define new table column types with fixed width.

```
\newcolumntype{L}[1]{>{\raggedright\let\newline\\%
\arraybackslash\hspace{0pt}}m{#1}}
\newcolumntype{C}[1]{>{\centering\let\newline\\%
\arraybackslash\hspace{0pt}}m{#1}}
\newcolumntype{R}[1]{>{\raggedleft\let\newline\\%
\arraybackslash\hspace{0pt}}m{#1}}
```

Misc and helper commands

Add comma in a cvReference entry.

```
\NewDocumentCommand{\cvComma}{}{,}
```

Define check if argument is empty argument order: `{argument}{if empty}{if not empty}`

Based on: http://tex.stackexchange.com/questions/63223

```
\ExplSyntaxOn
\DeclareExpandableDocumentCommand{\@IfNoValueOrEmptyTF}%
  {mmm}{\IfNoValueTF{#1}{#2}{
    \tl_if_empty:nTF {#1} {#2} {#3}%
   }
}
\ExplSyntaxOff
```

`\ifnodedefined` Check if node is defined. Based on: https://tex.stackexchange.com/questions/37709

```
\long\def\ifnodedefined#1#2#3{%
  \@ifundefined{pgf@sh@ns@#1}{#3}{#2}%
}
```

Update global font colour

```
\NewDocumentCommand{\globalcolor}{m}{%
  \color{#1}\global\let\default@color\current@color
}
\AtBeginDocument{\globalcolor{cvDark}}
```

TikZ

```
\RequirePackage{tikz}
\RequirePackage{graphicx}
\usetikzlibrary{%
  calc,
  positioning,
  fit,
  arrows.meta,
  backgrounds,
  matrix}
\tikzset{
  interesticon/.style = {},
  interesttext/.style = {},
  eventdottext/.style = {},
  invisibletimedot/.style = {},
  timedot/.style = {},
  contactIcon/.style={},
  contactText/.style={},
  headerIcon/.style={},
  skillLevel/.style={},
  languageText/.style={},
```

```
skillText/.style={},
progressArea/.style={},
progressBar/.style={},
sectionTitle/.style={},
sectionEduText/.style={%
eventdottext,
anchor=north west
},
cv@interesticon/.style = {%
  minimum width=\cvHeaderIconWidth,
  interesticon
},
cv@interesttext/.style = {%
  interesttext,
},
cv@eventdottext/.style = {%
  text width=\cvMainWidth-\cvTimeDotSep,
  black,
  anchor=north west,
  eventdottext,
},
cv@invisibletimedot/.style = {%
  circle,
  minimum width=\cvTimeDotDiameter,
  anchor=center,
  invisibletimedot,
},
cv@timedot/.style = {%
  cv@invisibletimedot,
  draw,
  fill,
  black,
  timedot,
},
cv@contactIcon/.style={%
  minimum height=\baselineskip,
  contactIcon,
},
cv@contactText/.style={%
  minimum height=\baselineskip,
  text depth=0pt,
  contactText,
},
cv@headerIcon/.style={%
  minimum width=\cvHeaderIconWidth,
  anchor=center,
  headerIcon,
},
cv@skillLevel/.style={skillLevel,},
cv@languageText/.style={languageText,},
```

```
  cv@skillText/.style={skillText,},
  cv@progressArea/.style={%
    minimum width=\cvProgressAreaWidth,
    minimum height=\cvProgressAreaHeight,
    rectangle,
    draw,
    cvGreen,
    progressArea
  },
  cv@progressBar/.style={%
    minimum height=\cvProgressAreaHeight,
    rectangle,
    draw,
    fill,
    cvGreen,
    anchor=west,
    progressBar
  },
  cv@sectionTitle/.style={%
    anchor=north west,
    align=left,
    sectionTitle,
  },
  cv@sectionEduText/.style={%
    cv@eventdottext,
    anchor=north west,
    sectionEduText,
  },
}
```

XeLaTeX/LuaLaTeX specific

```
\NewDocumentCommand{\cv@loadFonts}{}{
  \RequirePackage{fontspec}
  \RequirePackage{fontawesome}
```

set fonts

```
  \setmainfont[Numbers={Monospaced}]{\limecv@sansfont}
  \setsansfont{\limecv@sansfont}
  \setmonofont{\limecv@monofont}
  \ifcv@fontPath
    \newfontfamily{\cv@FA}[Path=\limecv@path]%
      {FontAwesome}
  \else
    \newfontfamily{\cv@FA}{FontAwesome}
  \fi
```

define some missing icons

```
  \ProvideDocumentCommand{\faAddressCard}{}{{\cv@FA%
    \symbol{"F2B9}}}
```

```
  \ProvideDocumentCommand{\faVcard}{}{{\cv@FA%
    \symbol{"F2BB}}}
  \ProvideDocumentCommand{\faMicrochip}{}{{\cv@FA%
    \symbol{"F2DB}}}
```

future possibility: https://raw.githubusercontent.com/xdanaux/moderncv/ master/moderncviconsmarvosym.sty

```
}
\ifxetex
  \cv@loadFonts
\else
  \ifluatex
    \cv@loadFonts
  \else
    \RequirePackage{fontawesome}
    \RenewDocumentCommand{\familydefault}{}{\sfdefault}
  \fi
\fi
```

XeLaTeX/LuaLaTeX specific
Removes left/right side bearing of character
Based on: https://tex.stackexchange.com/questions/274479 https://tex.stackexchange.com/questions/

```
\ifxetex
  \NewDocumentCommand{\kright}{m}{\leavevmode #1%
    \kern-\XeTeXglyphbounds3 \the\XeTeXcharglyph`#1 }
  \NewDocumentCommand{\kleft}{m}{\leavevmode %
    \kern-\XeTeXglyphbounds1 \the\XeTeXcharglyph`#1 #1}
\else
  \NewDocumentCommand{\kright}{m}{#1}
  \NewDocumentCommand{\kleft}{m}{#1}
\fi
```

Extracts n-th charcter from a string Argument order: string - character number

```
\ExplSyntaxOn
\cs_generate_variant:Nn\tl_item:nn {f}
\DeclareExpandableDocumentCommand{\extract}{mm}{%
  \tl_item:fn{#1}{#2}%
}
\ExplSyntaxOff
```

Page layout
remove page numbers

```
%% \pagenumbering{gobble}
```

remove all page styling

```
\pagestyle{empty}
```

not used

```
\newlength\cvColSep
\setlength\cvColSep{1cm} % 2 times column separation!
```

not used

```
\newlength\cvNodeSep
\setlength\cvNodeSep{0.75cm}
\newlength\cvTimeDotDiameter
\setlength\cvTimeDotDiameter{3pt}
\newlength\cvMargin
\setlength\cvMargin{1cm}
\newlength\cvSideWidth
\setlength\cvSideWidth{0.3\paperwidth-\cvMargin}
\newlength\cvMainWidth
\setlength\cvMainWidth{\paperwidth-4\cvMargin-\cvSideWidth}
\newlength\cvTimeDotSep
\setlength\cvTimeDotSep{0.4cm}
\newlength\cvStartEndSep
\setlength\cvStartEndSep{0.3cm}
\newlength\cvItemSep
\setlength\cvItemSep{0.4cm}
\newlength\cvTableSepWidth
\setlength\cvTableSepWidth{0.3cm}
\newlength\cvCoverLetterHeight
\setlength\cvCoverLetterHeight{\cvSideWidth}
\newlength\cvCoverLetterWidth
\setlength\cvCoverLetterWidth{\cvMainWidth+3\cvMargin}
\newlength\cvPictureWidth
\setlength\cvPictureWidth{4cm}
```

progress area size definition

```
\newlength\cvProgressAreaWidth
\setlength\cvProgressAreaWidth{5em}
\newlength\cvProgressAreaHeight
\setlength\cvProgressAreaHeight{0.75em}
\newlength\cvSectionSep
\setlength\cvSectionSep{0.6cm}
\newlength\cvSectionSBSep
\setlength\cvSectionSBSep{4pt}
```

line width of horizontal line below section title

```
\newlength\cvTitleLineWidth
\setlength\cvTitleLineWidth{2pt}
```

distance from bottom of text to centre of line

```
\newlength\cvTitleLineSpacing
\setlength\cvTitleLineSpacing{0.1cm}
```

set to `\cvItemSep` by default (do not overwrite!)

```
\newlength\cv@projectSep
\setlength\cv@projectSep{0cm}
```

internal spacing between details of a single project

```
\newlength\cvProjectDetailsSep
\setlength\cvProjectDetailsSep{0.2cm}
```

```
\newlength\cvInterestDetailsSep
\setlength\cvInterestDetailsSep{4pt}
\newlength\cvContactItemSep
\setlength\cvContactItemSep{5pt}
```

line width of line below cover letter name

```
\newlength\cvCoverLetterLineWidth
\setlength\cvCoverLetterLineWidth{3pt}
\newlength\cvCoverLetterPositionSpacing
\setlength\cvCoverLetterPositionSpacing{10pt}
\newlength\cvSBSectionLineWidth
\setlength\cvSBSectionLineWidth{1pt}
\newlength\cvCoverLetterLineSpacing
\setlength\cvCoverLetterLineSpacing{0pt}
\newlength\cvIDNameSep
\setlength\cvIDNameSep{0.1cm}
\newlength\cvPositionSep
\setlength\cvPositionSep{0.5cm}
\newlength\cvSkillSep
\setlength\cvSkillSep{1ex}
\newlength\cvHeaderIconWidth
\setlength\cvHeaderIconWidth{0pt}
```

New styling commands
Style main section title

```
\NewDocumentCommand{\cvSection}{m}{\Large\textbf{#1}}
```

Typeset skill level

```
\newcount\my@repeat@count
\NewDocumentCommand{\cvSkill}{m}{%
  \begingroup
  \my@repeat@count=\z@
  \@whilenum\my@repeat@count<#1\do{\faCircle\advance%
    \my@repeat@count\@ne}%
  \my@repeat@count=\numexpr5-\z@\relax
  \@whilenum\my@repeat@count>#1\do{\faCircleO\advance%
    \my@repeat@count\m@ne}%
  \endgroup
}
```

Create horizontal line from edge to edge for sidebar section titles.
Based on: https://tex.stackexchange.com/questions/65731 Alternative: https://tex.stackexchange.com

```
\NewDocumentCommand{\cvhrulefill}{}{{\color{cvGreen}%
\leavevmode\leaders\hrule height \cvSBSectionLineWidth\hfill\kern\z@}}
\NewDocumentCommand{\ruleline}{m}{\par\noindent\raisebox{.6ex}{%
\makebox[\linewidth]{\cvhrulefill\hspace{1ex}%
\raisebox{-.6ex}{#1}\hspace{1ex}\cvhrulefill}}}
```

Title counters

```
\newcounter{cv@titlePrev}
```

```
\setcounter{cv@titlePrev}{0}
\newcounter{cv@titleNext}
\setcounter{cv@titleNext}{1}
```

Draw everything associated with main section title.

```
\NewDocumentCommand{\cv@Title}{mm}{%
  \ifnodedefined{cv@last item \the\value{cv@lastItem}}{%
    \setlength\cvHeaderIconWidth{\maxof{\cvHeaderIconWidth}{\widthof{\Large #2}}}%
    \protected@write\@auxout{}{\global\cvHeaderIconWidth=\the\cvHeaderIconWidth}
    \node[%
      below=\cvSectionSep of cv@last item %
        \the\value{cv@lastItem}.south west,
      cv@sectionTitle]
      (cv@title \the\value{cv@titleNext})
      {\begingroup\edef\y{\endgroup%
      \cvSection\bgroup\noexpand\kleft{%
        \extract{#1}{1}}}%
      \y\StrGobbleLeft{#1}{1}\egroup};
    \node[%
      left=\cvTimeDotSep of cv@title \the\value{cv@titleNext},
      cv@headerIcon]
      {#2};
    \node[
      below=\cvSectionSep of cv@last item %
        \the\value{cv@lastItem}.south west,
      cv@sectionTitle]
      (cv@title \the\value{cv@titleNext} dummy) {%
        \phantom{\cvSection{Education}}};
    \begin{scope}[on background layer]
      \draw[line width=\cvTitleLineWidth,cvGreen]
        let \p1=(cv@title \the\value{cv@titleNext} %
                dummy.south west),
            \p2=(current page.east) in
        (\x1,\y1-\cvTitleLineSpacing) to (\x2,\y1-\cvTitleLineSpacing);
    \end{scope}
  }{%
    \node[cv@sectionTitle] at (0,0) (cv@title %
      \the\value{cv@titleNext})
    {\begingroup\edef\y{\endgroup%
    \cvSection\bgroup\noexpand\kleft{\extract{#1}{1}}}%
    \y\StrGobbleLeft{#1}{1}\egroup};
    \node[%
      left=\cvTimeDotSep of cv@title \the\value{cv@titleNext},
      cv@headerIcon]
      {#2};
    \node[cv@sectionTitle] at (0,0)
      (cv@title \the\value{cv@titleNext} dummy) {%
        \phantom{\cvSection{Education}}};
    \begin{scope}[on background layer]
      \draw[line width=\cvTitleLineWidth,cvGreen]
```

```
        let \p1=(cv@title \the\value{cv@titleNext} %
                dummy.south west),
            \p2=(current page.east) in
        (\x1,\y1-\cvTitleLineSpacing) to (\x2,\y1-\cvTitleLineSpacing);
    \end{scope}
}%
\stepcounter{cv@titlePrev}%
\stepcounter{cv@titleNext}
}
```

Draw the timeline

```
\NewDocumentCommand{\cv@EndSectionDraw}{}{
  \draw (start) to (end.center);}
```

Item counters

```
\newcounter{cv@itemPrev}
\setcounter{cv@itemPrev}{0}
\newcounter{cv@itemNext}
\setcounter{cv@itemNext}{1}
\newcounter{cv@lastItem}
\setcounter{cv@lastItem}{1}
```

List a timeline entry

```
\NewDocumentCommand{\cv@definecvItem}{}{
  \NewDocumentCommand{\cvItem}{m}{%
    \ifnodedefined{item \the\value{cv@itemPrev}}{%
      \node[%
        below=\cvItemSep of item %
          \the\value{cv@itemPrev}.south west,
        cv@eventdottext]
        (item \the\value{cv@itemNext} header)
        {\phantom{Evening}};
      \node[%
        below=\cvItemSep of item %
          \the\value{cv@itemPrev}.south west,
        cv@sectionEduText]
        (item \the\value{cv@itemNext})
        {##1};
      \node[%
        left=\cvTimeDotSep of item %
          \the\value{cv@itemNext} header,
        cv@timedot]
        {};
    }{
      \node[%
        below=\cvItemSep of cv@title %
          \the\value{cv@titlePrev}.south west,
        cv@eventdottext]
        (item \the\value{cv@itemNext} header)
        {\phantom{Evening}};
      \node[
```

```
        below=\cvItemSep of cv@title %
          \the\value{cv@titlePrev}.south west,
        cv@sectionEduText]
        (item \the\value{cv@itemNext})
        {##1};
      \node[
        left=\cvTimeDotSep of item %
          \the\value{cv@itemNext} header,
        cv@timedot]
        (start)
        {};
    }
    \node[
      left=\cvTimeDotSep of item %
        \the\value{cv@itemNext}.south west,
      cv@invisibletimedot]
      (end)
      {};
    \stepcounter{cv@itemPrev}
    \stepcounter{cv@itemNext}
  }
}
```

——————— Language

```
\pgfkeys{/@cv/names/.is family,
         /@cv/names/.unknown/.style = %
         {\pgfkeyscurrentpath/\pgfkeyscurrentname/%
         .initial = #1}}
```

Set all title names to English (default):

```
\NewDocumentCommand{\cv@setEnglish}{}{
    \pgfkeys{/@cv/names/profile = Profile}%
    \pgfkeys{/@cv/names/contact = Contact}%
    \pgfkeys{/@cv/names/languages = Languages}%
    \pgfkeys{/@cv/names/interests = Interests}%
    \pgfkeys{/@cv/names/professional = Professional}%
    \pgfkeys{/@cv/names/personal = Personal}%
    \pgfkeys{/@cv/names/projects = Projects}%
    \pgfkeys{/@cv/names/education = Education}%
    \pgfkeys{/@cv/names/experience = Experience}%
    \pgfkeys{/@cv/names/references = References}%
    \pgfkeys{/@cv/names/skills = Skills}%
}
```

Set all title names to Dutch:

```
\NewDocumentCommand{\cv@setDutch}{}{
    \pgfkeys{/@cv/names/profile = Profiel}%
    \pgfkeys{/@cv/names/contact = Gegevens}%
    \pgfkeys{/@cv/names/languages = Talenkennis}%
    \pgfkeys{/@cv/names/interests = Interesses}%
    \pgfkeys{/@cv/names/professional = Professioneel}%
```

```
    \pgfkeys{/@cv/names/personal = Persoonlijk}%
    \pgfkeys{/@cv/names/projects = Projecten}%
    \pgfkeys{/@cv/names/education = Opleiding}%
    \pgfkeys{/@cv/names/experience = Werkervaring}%
    \pgfkeys{/@cv/names/references = Referenties}%
    \pgfkeys{/@cv/names/skills = Vaardigheden}%
}
```

Set the actual language to English. This can be overwritten by the user in the preamble

Define the command to change the language

```
\ExplSyntaxOn
\NewDocumentCommand{\cvSetLanguage}{m}{
    \str_case:nnF {#1}
    {
      {dutch} {\cv@setDutch}
      {english} {\cv@setEnglish}
    }
    {\ClassError{limecv}{Unknown option ‘#1’ for cvLanguage %
     macro.}{Only ‘english’ and ‘dutch’ are valid options %
     for this macro.}}
}
\ExplSyntaxOff
\cvSetLanguage{english}
% \cvSetLanguage{english}
```

Commands specific for cvSkills section
information in temporary command yes/no

```
\newif\ifcv@firstskill
```

temporary command that stores skill information

```
\NewDocumentCommand{\cv@skillsTable}{}{}
```

typeset two skills

```
\NewDocumentCommand{\cvSkillTwo}{mmmm}{%
  \ifcv@firstskill
    \g@addto@macro\cv@skillsTable{\cvSkill{#1} %
      & #2 & \cvSkill{#3} & #4}
    \cv@firstskillfalse
  \else
    \g@addto@macro\cv@skillsTable{\\[\cvSkillSep] \cvSkill{#1} %
      & #2 & \cvSkill{#3} & #4}
  \fi
}
```

typeset one skill

```
\NewDocumentCommand{\cvSkillOne}{mm}{%
  \ifcv@firstskill
    \g@addto@macro\cv@skillsTable{\cvSkill{#1} & #2}
    \cv@firstskillfalse
```

```
  \else
    \g@addto@macro\cv@skillsTable{\\[\cvSkillSep] \cvSkill{#1} & #2}
  \fi
}
```

---

cvMainContent environments

```
\NewDocumentEnvironment{cvMainContent}{}{%
  \begin{tikzpicture}[%
    every node/.style={%
      inner sep=0pt,
      outer sep=0pt},
    remember picture,
    overlay,
    shift={($(current page.north west)+%
      (\cvSideWidth+3\cvMargin+\cvTimeDotSep,-\cvMargin)$)}]
}{%
  \end{tikzpicture}
}
\NewDocumentEnvironment{cvMainContent*}{}{%
  \stepcounter{titlePrev}
  \stepcounter{titleNext}
    \begin{tikzpicture}[%
      every node/.style={
        inner sep=0pt,
        outer sep=0pt},
      remember picture,
      overlay,
      shift={($(current page.north west)+%
      (\cvMargin+\cvTimeDotSep,-\cvMargin)$)}]
}{%
    \end{tikzpicture}%
}
```

---

cvEductation environment

```
\NewDocumentEnvironment{cvEducation}{}{%
    \cv@Title{\pgfkeysvalueof{/@cv/names/education}}{\faGraduationCap}
```

insures uniqueness

```
    \stepcounter{cv@itemPrev}
    \stepcounter{cv@itemNext}
    \cv@definecvItem
  }{%
    \cv@EndSectionDraw
    \stepcounter{cv@lastItem}
    \draw (item \the\value{cv@itemPrev}.south west)
      node (cv@last item \the\value{cv@lastItem}) {};
}
```

cvExperience environment

```
\NewDocumentEnvironment{cvExperience}{}{%
  \cv@Title{\pgfkeysvalueof{/@cv/names/experience}}{\faBriefcase}
```

insures uniqueness

```
  \stepcounter{cv@itemPrev}
  \stepcounter{cv@itemNext}
  \cv@definecvItem
}{%
  \cv@EndSectionDraw
  \stepcounter{cv@lastItem}
  \draw (item \the\value{cv@itemPrev}.south west)
    node (cv@last item \the\value{cv@lastItem}) {};
}
```

cvSkills environment

```
\NewDocumentEnvironment{cvSkills}{}{%
  \cv@Title{\pgfkeysvalueof{/@cv/names/skills}}{\faStar}
  \global\cv@firstskilltrue
}{%
  \node[
    below=\cvSectionSep of cv@title \the\value{cv@titlePrev}%
      .south west,
    anchor=north west] (skills) %
    {\begin{tabular}{@{}ll@{\hskip\cvColSep}ll@{}}%
    \cv@skillsTable
    \end{tabular}};
  \stepcounter{cv@lastItem}
  \draw (skills.south west) node %
    (cv@last item \the\value{cv@lastItem}) {};
}
```

cvReferences environment

```
\NewDocumentEnvironment{cvReferences}{}{%
  \cv@Title{\pgfkeysvalueof{/@cv/names/references}}{\faLink}
  \node[
    below=\cvSectionSep of cv@title \the\value{cv@titlePrev}.%
      south west,
    anchor=north west,
    ] (cv@References) \bgroup %
}{%
  \egroup;
  \stepcounter{cv@lastItem}
  \draw (cv@References.south west) node %
    (cv@last item \the\value{cv@lastItem}) {};
}
```

Key-value parsing structure

```
\pgfkeys{/@cv/references/.is family,
         /@cv/references/.unknown/.style = %
         {\pgfkeyscurrentpath/\pgfkeyscurrentname/%
         .initial = #1}}
\newif\ifcv@refName
\pgfkeys{/@cv/references/name/.is if=cv@refName}
\pgfkeys{/@cv/references/name=false}
\newif\ifcv@refPosition
\pgfkeys{/@cv/references/position/.is if=cv@refPosition}
\pgfkeys{/@cv/references/position=false}
\newif\ifcv@refCompany
\pgfkeys{/@cv/references/company/.is if=cv@refCompany}
\pgfkeys{/@cv/references/copmany=false}
\newif\ifcv@refAddressLineA
\pgfkeys{/@cv/references/address line 1/.is if=%
  cv@refAddressLineA}
\pgfkeys{/@cv/references/address line 1=false}
  \newif\ifcv@refAddressLineB
\pgfkeys{/@cv/references/address line 2/.is if=%
  cv@refAddressLineB}
\pgfkeys{/@cv/references/address line 2=false}
\newif\ifcv@refAddressLineC
\pgfkeys{/@cv/references/address line 3/.is if=%
  cv@refAddressLineC}
\pgfkeys{/@cv/references/address line 3=false}
\newif\ifcv@refWorkPhone
\pgfkeys{/@cv/references/work phone/.is if=%
  cv@refWorkPhone}
\pgfkeys{/@cv/references/work phone=false}
\newif\ifcv@refMobilePhone
\pgfkeys{/@cv/references/mobile phone/.is if=%
  cv@refMobilePhone}
\pgfkeys{/@cv/references/mobile phone=false}
\newif\ifcv@refStored
\pgfkeys{/@cv/references/value stored/.is if=%
  cv@refStored}
\pgfkeys{/@cv/references/value stored=false}
```

keys

```
\define@key{cvAddReferenceA}{name}{%
  \global\def\cv@refAName{#1}%
  \@IfNoValueOrEmptyTF{#1}{}{\pgfkeys{/@cv/references/name = true}}}
\define@key{cvAddReferenceA}{company}{%
  \global\def\cv@refACompany{#1}%
  \@IfNoValueOrEmptyTF{#1}{}{\pgfkeys{/@cv/references/company = true}}}
\define@key{cvAddReferenceA}{position}{%
  \global\def\cv@refAPosition{#1}%
  \@IfNoValueOrEmptyTF{#1}{}{\pgfkeys{/@cv/references/position = true}}}
```

```
\define@key{cvAddReferenceA}{address line 1}{%
  \global\def\cv@refAAddressLineA{#1}%
  \@IfNoValueOrEmptyTF{#1}{}{\pgfkeys{/@cv/references/address line 1 = true}}}
\define@key{cvAddReferenceA}{address line 2}{%
  \global\def\cv@refAAddressLineB{#1}%
  \@IfNoValueOrEmptyTF{#1}{}{\pgfkeys{/@cv/references/address line 2 = true}}}
\define@key{cvAddReferenceA}{address line 3}{%
  \global\def\cv@refAAddressLineC{#1}%
  \@IfNoValueOrEmptyTF{#1}{}{\pgfkeys{/@cv/references/address line 3 = true}}}
\define@key{cvAddReferenceA}{phone work}{%
  \global\def\cv@refAPhoneWork{#1}%
  \@IfNoValueOrEmptyTF{#1}{}{\pgfkeys{/@cv/references/phone work = true}}}
\define@key{cvAddReferenceA}{mobile phone}{%
  \global\def\cv@refAMobilePhone{#1}%
  \@IfNoValueOrEmptyTF{#1}{}{\pgfkeys{/@cv/references/mobile phone = true}}}
\define@key{cvAddReferenceB}{name}{%
  \global\def\cv@refBName{#1}%
  \@IfNoValueOrEmptyTF{#1}{}{\pgfkeys{/@cv/references/name = true}}}
\define@key{cvAddReferenceB}{company}{%
  \global\def\cv@refBCompany{#1}%
  \@IfNoValueOrEmptyTF{#1}{}{\pgfkeys{/@cv/references/company = true}}}
\define@key{cvAddReferenceB}{position}{%
  \global\def\cv@refBPosition{#1}%
  \@IfNoValueOrEmptyTF{#1}{}{\pgfkeys{/@cv/references/position = true}}}
\define@key{cvAddReferenceB}{address line 1}{%
  \global\def\cv@refBAddressLineA{#1}%
  \@IfNoValueOrEmptyTF{#1}{}{\pgfkeys{/@cv/references/address line 1 = true}}}
\define@key{cvAddReferenceB}{address line 2}{%
  \global\def\cv@refBAddressLineB{#1}%
  \@IfNoValueOrEmptyTF{#1}{}{\pgfkeys{/@cv/references/address line 2 = true}}}
\define@key{cvAddReferenceB}{address line 3}{%
  \global\def\cv@refBAddressLineC{#1}%
  \@IfNoValueOrEmptyTF{#1}{}{\pgfkeys{/@cv/references/address line 3 = true}}}
\define@key{cvAddReferenceB}{phone work}{%
  \global\def\cv@refBPhoneWork{#1}%
  \@IfNoValueOrEmptyTF{#1}{}{\pgfkeys{/@cv/references/phone work = true}}}
\define@key{cvAddReferenceB}{mobile phone}{%
  \global\def\cv@refBMobilePhone{#1}%
  \@IfNoValueOrEmptyTF{#1}{}{\pgfkeys{/@cv/references/mobile phone = true}}}
\NewDocumentCommand{\cvAddReference}{sm}{%
  \IfBooleanTF{#1}{%
    \setkeys{cvAddReferenceA}{name=,company=,
      position=,address line 1=,address line 2=,
      address line 3=,phone work=,mobile phone=,#2}
    \pgfkeys{/@cv/references/name = true}%
```

```
  }{%
    \setkeys{cvAddReferenceB}{name=,company=,
    position=,address line 1=,address line 2=,
    address line 3=,phone work=,mobile phone=,#2}
    \ifcv@refStored%
      \begin{tabular}{@{}L{0.5\cvMainWidth-0.5\cvTimeDotSep%
            -0.5\cvTableSepWidth}@{\hskip \cvTableSepWidth}%
          L{0.5\cvMainWidth-0.5\cvTimeDotSep%
            -0.5\cvTableSepWidth}@{}}%
        \ifcv@refName            \cv@refAName         & \cv@refBName         \fi %
        \ifcv@refCompany      \\ \cv@refACompany      & \cv@refBCompany      \fi %
        \ifcv@refPosition     \\ \cv@refAPosition     & \cv@refBPosition     \fi %
        \ifcv@refAddressLineA \\ \cv@refAAddressLineA & \cv@refBAddressLineA \fi %
        \ifcv@refAddressLineB \\ \cv@refAAddressLineB & \cv@refBAddressLineB \fi %
        \ifcv@refAddressLineC \\ \cv@refAAddressLineC & \cv@refBAddressLineC \fi %
        \ifcv@refWorkPhone    \\ \cv@refAPhoneWork    & \cv@refBPhoneWork    \fi %
        \ifcv@refMobilePhone  \\ \cv@refAMobilePhone  & \cv@refBMobilePhone  \fi %
      \end{tabular}%
      \pgfkeys{/@cv/references/value stored = false}%
    \else%
      \begin{tabular}{@{}L{0.5\cvMainWidth%
          -0.5\cvTimeDotSep}@{}}%
        \ifcv@refName            \cv@refBName         \fi %
        \ifcv@refCompany      \\ \cv@refBCompany      \fi %
        \ifcv@refPosition     \\ \cv@refBPosition     \fi %
        \ifcv@refAddressLineA \\ \cv@refBAddressLineA \fi %
        \ifcv@refAddressLineB \\ \cv@refBAddressLineB \fi %
        \ifcv@refAddressLineC \\ \cv@refBAddressLineC \fi %
        \ifcv@refWorkPhone    \\ \cv@refBPhoneWork    \fi %
        \ifcv@refMobilePhone  \\ \cv@refBMobilePhone  \fi %
      \end{tabular}%
    \fi%
  }%
}
```

---

cvSidebar environment

```
\NewDocumentEnvironment{cvSidebar}{}{%
  \vspace*{\fill}
  \vspace*{-0.5\cvMargin}
  \begin{tikzpicture}[remember picture,overlay]
    \fill[cvGreenLight] (current page.north west) rectangle %
      ++(\cvSideWidth+2\cvMargin,-\paperheight);
  \end{tikzpicture}%
  \begin{minipage}{\cvSideWidth}
    \begin{center}
}{%
    \end{center}
  \end{minipage}
  \vspace*{\fill}
```

```
}
\NewDocumentEnvironment{cvSidear*}{}{%
  \vspace*{\fill}
  \vspace*{-0.5\cvMargin}
  \begin{tikzpicture}[remember picture,overlay]
    \fill[cvGreenLight] (current page.north east) rectangle %
      ++(-\cvSideWidth-2\cvMargin,-\paperheight);
  \end{tikzpicture}%
  \begin{minipage}{\cvSideWidth}
    \begin{center}
}{%
    \end{center}
  \end{minipage}
  \vspace*{\fill}
}
```

---

cvProfile environment

```
\NewDocumentEnvironment{cvProfile}{}{%
  \vspace{\cvSectionSBSep}
  \ruleline{\pgfkeysvalueof{/@cv/names/profile}}
}{%
}
```

---

cvContact environment

```
\NewDocumentCommand{\cv@ContactTemplate}{mm}{
  \NewDocumentCommand{#1}{m}{
      \ifcv@contactfirst \\ \else %
        \global\cv@contactfirsttrue \fi\node[cv@contactIcon]{#2};
        \pgfmatrixnextcell \node[cv@contactText]{##1};
  }
}
\NewDocumentCommand{\cv@ContactTemplateLink}{mm}{
  \NewDocumentCommand{#1}{mm}{
      \ifcv@contactfirst \\ \else %
        \global\cv@contactfirsttrue \fi\node[cv@contactIcon]{#2};
        \pgfmatrixnextcell \node[cv@contactText]{%
          \@IfNoValueOrEmptyTF{##1}{##2}{\href{##1}{##2}}};
  }
}
\newif\ifcv@contactfirst
\NewDocumentEnvironment{cvContact}{}{%
  \vspace{\cvSectionSBSep}
  \ruleline{\pgfkeysvalueof{/@cv/names/contact}}%
  \vspace{\cvSectionSBSep}

  \global\cv@contactfirstfalse
  \cv@ContactTemplate{\cvContactAddress}{\faMapMarker}
  \cv@ContactTemplate{\cvContactPhone}{\faPhone}
```

```
  \cv@ContactTemplateLink{\cvContactEmail}{\faEnvelope}
  \cv@ContactTemplateLink{\cvContactWebsite}{\faGlobe}
  \cv@ContactTemplateLink{\cvContactGithub}{\faGithub}
  \cv@ContactTemplateLink{\cvContactLinkedin}{\faLinkedinSquare}
  \cv@ContactTemplateLink{\cvContactTwitter}{\faTwitter}
  \cv@ContactTemplateLink{\cvContactKeybase}{\faKey}
  \begin{tikzpicture}[%
    every node/.style={inner sep=0pt, outer sep=0pt}]
    \matrix [
      column 1/.style={anchor=center},
      column 2/.style={anchor=west,align=left},
      column sep=\cvContactItemSep,
      row sep=\cvContactItemSep,
      inner sep=0pt,
      outer sep=0pt] (contact) \bgroup
}{%
  \\\egroup; % \\ required, does not compile otherwise
  \end{tikzpicture}%
}
```

---

cvInterests environment

```
\newif\ifcv@interestLong
\cv@interestLongtrue
\newcounter{cv@cvInterestCounter}
\setcounter{cv@cvInterestCounter}{1}
\newcounter{cv@cvInterestCounterPrev}
\setcounter{cv@cvInterestCounterPrev}{0}
\newif\ifcv@interestContent
\cv@interestContentfalse
\NewDocumentCommand{\cv@TypesetTableRows}{}{}
\newif\ifcv@interesttikzopen
\cv@interesttikzopenfalse
\NewDocumentCommand{\cv@interestClose}{}{
  \ifcv@interestLong
    \ifcv@interesttikzopen
      \end{tikzpicture}
      \global\cv@interesttikzopenfalse
    \fi
  \else
    \ifcv@interestContent
      \begin{tabular}{@{}cl@{}}
        \cv@TypesetTableRows
      \end{tabular}
    \fi
  \fi
}
\NewDocumentCommand{\cv@interestOpen}{}{
  \ifcv@interestLong
```

```
    \ifcv@interesttikzopen
    \else
      \cv@interesttikzopentrue
      \begin{tikzpicture}[%
        every node/.style={inner sep=0cm,outer sep=0cm}]
    \fi
  \else
    \global\def\cv@TypesetTableRows{}
    \global\cv@interestContentfalse
  \fi
}
\NewDocumentCommand{\cv@definecv@interestSection}{}{
  \NewDocumentCommand{\cv@interestSection}{m}{%
    \cv@interestClose\\
    \ifcv@interestContent
      \vspace*{2\cvSectionSBSep}
    \fi
    {\color{cvGreen}---} ##1 {\color{cvGreen}---}\\
    \vspace*{\cvSectionSBSep}
    \cv@interestOpen
  }
}
\NewDocumentCommand{\cv@definecvInterestLong}{}{
  \NewDocumentCommand{\cvInterest}{mm}{%
    \ifcv@interestContent %
      \node[
        below=\cvItemSep of cv@interest text \the\value{cv@cvInterestCounterPrev},
        cv@interesticon]
        (cv@interest icon \the\value{cv@cvInterestCounter}) {##1};
      \node[
        below=\cvInterestDetailsSep of cv@interest icon \the\value{cv@cvInterestCounter},
        cv@interesttext]
      (cv@interest text \the\value{cv@cvInterestCounter}) {##2};
    \else
      \cv@interestOpen
      \node[cv@interesticon] (cv@interest icon \the\value{cv@cvInterestCounter}) {##1};
      \node[
        below=\cvInterestDetailsSep of cv@interest icon \the\value{cv@cvInterestCounter},
        cv@interesttext]
        (cv@interest text \the\value{cv@cvInterestCounter}) {##2};
    \fi
    \stepcounter{cv@cvInterestCounter}
    \stepcounter{cv@cvInterestCounterPrev}
    \global\cv@interestContenttrue
  }
}
\NewDocumentCommand{\cv@definecvInterestShort}{}{
```

https://tex.stackexchange.com/questions/12761

```
  %\def\Z{\vphantom{\parbox[c]{1cm}{\Huge Something Long}}}
```

```
  \NewDocumentCommand{\cvInterest}{mm}{%
```
parse optinal argument if present
```
    \g@addto@macro\cv@TypesetTableRows{##1}%
    \g@addto@macro\cv@TypesetTableRows{&}%
    \g@addto@macro\cv@TypesetTableRows{##2}%
    \g@addto@macro\cv@TypesetTableRows{\\[\cvInterestDetailsSep]}%
    \global\cv@interestContenttrue
  }
}
\ExplSyntaxOn
\NewDocumentEnvironment{cvInterests}{o}{%
```
set defaults
```
  \cv@interestLongtrue
```
parse optional arguments
```

  \@IfNoValueOrEmptyTF{#1}{}{%
    \str_case:nnF {#1}{
      {long}{\cv@interestLongtrue}
      {short}{\cv@interestLongfalse}
    }{\ClassError{limecv}{Unkown option ‘#1’ for cvInterests %
      environment.}{Only ‘long’ and ‘short’ are valid options %
      for this environment.}}
  }

  \cv@definecv@interestSection
```
define new commands
```
  \NewDocumentCommand{\cvInterestsPersonal}{}{%
    \cv@interestSection{\faUser\ \emph{Personal}}}
  \NewDocumentCommand{\cvInterestsProfessional}{}{%
    \cv@interestSection{\faBriefcase\ \emph{Professional}}}
  \ifcv@interestLong
    \cv@definecvInterestLong
  \else
    \cv@definecvInterestShort
  \fi
  %%%%%%
  \vspace{\cvSectionSBSep}
  \ruleline{\pgfkeysvalueof{/@cv/names/interests}}%
  \vspace{\cvSectionSBSep}
}{%
  \cv@interestClose
}
\ExplSyntaxOff
```

---

cvProject environment

```
\newif\ifcv@projectImage
\cv@projectImagefalse
\newif\ifcv@projectLink
\cv@projectLinkfalse
\newif\ifcv@projectWidth
\cv@projectWidthfalse
\newif\ifcv@projectHeight
\cv@projectHeightfalse
\newcounter{cv@projectCurrent}
\setcounter{cv@projectCurrent}{1}
\newcounter{cv@projectPrevious}
\setcounter{cv@projectPrevious}{0}
\define@key{cv@Project}{image}{%
  \global\def\cv@projectImage{#1}%
  \cv@projectImagetrue}
\define@key{cv@Project}{link}{%
  \global\def\cv@projectLink{#1}%
  \cv@projectLinktrue}
\define@key{cv@Project}{width}{%
  \global\def\cv@projectWidth{#1}%
  \cv@projectWidthtrue}
\define@key{cv@Project}{height}{%
  \global\def\cv@projectHeight{#1}%
  \cv@projectHeighttrue}
\NewDocumentCommand{\cv@definecvProject}{}{
  \NewDocumentCommand{\cvProject}{omm}{%
```

default settings

```
    \cv@projectImagefalse
    \cv@projectLinkfalse
    \cv@projectWidthfalse
    \cv@projectHeightfalse
```

parse optional arguments

```
    \@IfNoValueOrEmptyTF{##1}{}{\setkeys{cv@Project}{##1}}
    \ifnodedefined{project \the\value{cv@projectPrevious} %
      description}{}{%
      \node (project \the\value{cv@projectPrevious} %
        description) {};
    }
```

draw image

```
    \ifcv@projectImage
      \ifcv@projectWidth
        \node[below=\cv@projectSep of project %
          \the\value{cv@projectPrevious} description]
          (project \the\value{cv@projectCurrent} image)
          {\includegraphics[width=\cv@projectWidth]{\cv@projectImage}};
      \else
```

```
        \ifcv@projectHeight
          \node[below=\cv@projectSep of project %
          \the\value{cv@projectPrevious} description]
          (project \the\value{cv@projectCurrent} image)
          {\includegraphics[height=\cv@projectHeight]{\cv@projectImage}};
        \else
          \ClassError{limecv}{No width or height specified, %
          but image specified.}{Always specify the height or %
          width if an image is set.}
        \fi
      \fi
```

draw title (and add href if specified)

```
      \node[%
        text width=\cvSideWidth,
        align=center,
        below=\cvProjectDetailsSep of project \the\value{cv@projectCurrent} image]
        (project \the\value{cv@projectCurrent} title) {%
      \ifcv@projectLink \href{\cv@projectLink}{##2} \else ##2 \fi };
    \else
```

draw title (and add href if specified)

```
      \node[below=\cv@projectSep of project %
        \the\value{cv@projectPrevious} description,
        text width=\cvSideWidth,
        align=center]
        (project \the\value{cv@projectCurrent} title) {%
      \ifcv@projectLink \href{\cv@projectLink}{##2} \else ##2 \fi };
    \fi
    \node[
      below=\cvProjectDetailsSep of project \the\value{cv@projectCurrent} title,
      text width=\cvSideWidth,
      align=center]
      (project \the\value{cv@projectCurrent} description) {##3};

    \setlength\cv@projectSep\cvItemSep
```

increase counter value

```
    \stepcounter{cv@projectCurrent}
    \stepcounter{cv@projectPrevious}
  }
}
\NewDocumentEnvironment{cvProjects}{}{%
  \vspace{\cvSectionSBSep}
  \ruleline{\pgfkeysvalueof{/@cv/names/projects}}%
  \vspace{\cvSectionSBSep}
  \cv@definecvProject
  \begin{tikzpicture}[every node/.style={inner sep=0cm,outer sep=0cm}]
}{
  \end{tikzpicture}
}
```

cvLanguages environment

```
\newcounter{cv@languages}
\setcounter{cv@languages}{1}
\pgfkeys{/@cv/languages/.is family,
/@cv/languages/.unknown/.style = {%
  \pgfkeyscurrentpath/\pgfkeyscurrentname/.initial = #1}}
\NewDocumentEnvironment{cvLanguages}{}{%
  \NewDocumentCommand{\cvLanguage}{mm}{%
    {\globaldefs=1\relax\pgfkeys{%
      /@cv/languages/lang\the\value{cv@languages} = ##2}}
    \node[cv@languageText]{##1}; \pgfmatrixnextcell %
      \node[cv@progressArea] (cv@language \the\value{cv@languages}) {}; \\
    \stepcounter{cv@languages}
  }%
  \vspace{\cvSectionSBSep}
  \ruleline{\pgfkeysvalueof{/@cv/names/languages}}%
  \vspace{\cvSectionSBSep}
  \begin{tikzpicture}[%
    every node/.style={text depth=0pt,inner sep=0pt,outer sep=0pt}]
  \matrix [
    column 1/.style={anchor=east},
    column sep=1.5\cvSectionSBSep,
    row sep=1.5\cvSectionSBSep,
    column 2/.style={anchor=west,align=left}
  ] (contact) \bgroup
}{%
  \\\egroup;
  \newcounter{cv@languagelevel}
  \setcounter{cv@languagelevel}{1}
  \loop\ifnum\value{cv@languagelevel}<\value{cv@languages}
    \draw (cv@language \the\value{cv@languagelevel}.west) %
      node[%
        cv@progressBar,
        minimum width=\pgfkeysvalueof{%
        /@cv/languages/lang\the\value{cv@languagelevel}}em]%
        {};
    \stepcounter{cv@languagelevel}
  \repeat
  \end{tikzpicture}
}
```

cvCoverLetter environment
cover letter keys

```
\define@key{cvCoverLetter}{name}{%
```

`\cv@coverletterName`

```
\global\def\cv@coverletterName{#1}}
```

```
\define@key{cvCoverLetter}{company}{%
\cv@coverletterCompany
\global\def\cv@coverletterCompany{#1}}
\define@key{cvCoverLetter}{position}{%
\cv@coverletterPosition
\global\def\cv@coverletterPosition{#1}}
\define@key{cvCoverLetter}{address line 1}{%
\cv@coverletterAddressLineA
\global\def\cv@coverletterAddressLineA{#1}}
\define@key{cvCoverLetter}{address line 2}{%
\cv@coverletterAddressLineB
\global\def\cv@coverletterAddressLineB{#1}}
\define@key{cvCoverLetter}{address line 3}{%
\cv@coverletterAddressLineC
\global\def\cv@coverletterAddressLineC{#1}}
\NewDocumentCommand{\cv@definecvBeneficiary}{}{
  \NewDocumentCommand{\cvBeneficiary}{m}{%
    \setkeys{cvCoverLetter}{name={},company={},
      position={},address line 1={},address line 2={},
      address line 3={},##1}
    \begin{tabular}{@{}l}
      \@IfNoValueOrEmptyTF{%
        \cv@coverletterName}{}{%
        \MakeUppercase{\cv@coverletterName}} %
      \@IfNoValueOrEmptyTF{%
        \cv@coverletterCompany}{}{\\ %
        \cv@coverletterCompany} %
      \@IfNoValueOrEmptyTF{%
        \cv@coverletterPosition}{}{\\ %
        \cv@coverletterPosition} %
      \@IfNoValueOrEmptyTF{%
        \cv@coverletterAddressLineA}{}{\\ %
        \cv@coverletterAddressLineA} %
      \@IfNoValueOrEmptyTF{%
        \cv@coverletterAddressLineB}{}{\\ %
        \cv@coverletterAddressLineB} %
      \@IfNoValueOrEmptyTF{%
        \cv@coverletterAddressLineC}{}{\\ %
        \cv@coverletterAddressLineC} %
    \end{tabular}%
    \vspace{\cvMargin}
  }
}
```

```
\NewDocumentCommand{\cv@definecvFullName}{}{
  \NewDocumentCommand{\cvFullName}{}{%
    \pgfkeysvalueof{/@cv/info/first name} %
    \pgfkeysvalueof{/@cv/info/last name}
  }
}
\NewDocumentEnvironment{cvCoverLetter}{}{%
  \newgeometry{textwidth=\cvCoverLetterWidth,noheadfoot}
  \begin{tikzpicture}[%
      remember picture,
      overlay,
      every node/.style={inner sep=0cm,outer sep=0cm}]
```

draw green background

```
    \begin{scope}[on background layer]
      \fill[cvGreenLight] (current page.north west) rectangle %
        ++(\paperwidth,-\cvCoverLetterHeight);
    \end{scope}
    \draw (current page.north east) %
      ++(-0.5\paperwidth+0.5\cvCoverLetterWidth,
         -\cvCoverLetterHeight/2) node (cv@h7) {};
    \node[anchor=east] at (cv@h7) (cv cover letter name){%
      \fontsize{50}{60}\selectfont
```

trick to expand argument such that ‘kleft‘ sees an alphabetic constant (i.e. a single letter)

```
      \begingroup\edef\y{\endgroup%
        \noexpand\kleft{\cv@FirstNameLetter}}%
      \y
      \StrGobbleLeft{\pgfkeysvalueof{/@cv/info/first name}}{1}\space
      \StrGobbleRight{\pgfkeysvalueof{/@cv/info/last name}}{1}
```

same trick trick to expand argument such that ‘kleft‘ sees an alphabetic constant (i.e. a single letter)

```
      \begingroup\edef\y{\endgroup%
        \noexpand\kright{\cv@LastNameLetter}}%
      \y
      };
    \node[%
      align=right,
      cvAccent,
      below=\cvCoverLetterPositionSpacing of cv cover letter name.south east,
      anchor=north east]{%
        \StrGobbleRight{\pgfkeysvalueof{/@cv/info/position}}{1}
    \begingroup\edef\y{\endgroup%
      \noexpand\kright{\cv@PositionLetter}}%
    \y};

    \begin{scope}[on background layer]
      \draw[line width=\cvCoverLetterLineWidth,cvGreen]
         ($(cv cover letter name.south west)+(0,-\cvCoverLetterLineSpacing)$) to
```

```
        ($(cv cover letter name.south east)+(0,-\cvCoverLetterLineSpacing)$);
    \end{scope}
  \end{tikzpicture}
  \vspace{\dimexpr\cvCoverLetterHeight-1in-\topmargin\relax}

  \today

  \vspace{\baselineskip}
  \cv@definecvBeneficiary
  \cv@definecvFullName
}{%
  \restoregeometry
}
```

---

cvID macro

```
\pgfkeys{/@cv/info/.is family,
        /@cv/info/.unknown/.style = %
        {\pgfkeyscurrentpath/\pgfkeyscurrentname/%
        .initial = #1}}
```

first name, second name, picture, position

```
\NewDocumentCommand{\cvID}{mmmm}{
  \@IfNoValueOrEmptyTF{#3}{}{
    \begin{tikzpicture}
      \node[
        circle,
        text=white,
        minimum size=\cvPictureWidth,
        path picture={
          \node at (path picture bounding box.center){
            \includegraphics[width=\cvPictureWidth]{#3}
          };
        }]
        {};
    \end{tikzpicture}\\
  }
  {\LARGE
  \vspace{\cvIDNameSep}
  #1
  \vspace{\cvIDNameSep}
  #2}

  \vspace{\cvPositionSep}

  \@IfNoValueOrEmptyTF{#4}{}{\color{cvAccent} #4}

  \vspace{\cvPositionSep}
  {\globaldefs=1\relax\pgfkeys{/@cv/info/first name = %
    #1}\def\cv@FirstNameLetter{\extract{#1}{1}}}
```

```
  {\globaldefs=1\relax\pgfkeys{/@cv/info/last name = %
    #2}\def\cv@LastNameLetter{\extract{#2}{-1}}}
  {\globaldefs=1\relax\pgfkeys{/@cv/info/position = %
    #4}\def\cv@PositionLetter{\extract{#4}{-1}}}
}
```