Code Listing Generator Generated Code Listing

bkettle

September 24, 2020

0.1 template.tex

```
\documentclass[12pt]{article}
\usepackage[utf8]{inputenc}
\usepackage[margin=1in]{geometry}
\usepackage{enumitem}
\usepackage{listings}
\usepackage{fancyhdr}
\usepackage{color}
\usepackage{courier}
\usepackage{multicol}
\usepackage{graphicx}
\usepackage{times}
\usepackage[%
    colorlinks=true,
    pdfborder={0 0 0},
    linkcolor=blue
]{hyperref}
\usepackage[all]{hypcap}
\usepackage{float}
\usepackage{textcomp}
\usepackage[superscript,biblabel]{cite}
\usepackage{placeins}

\title{\vspace{-2cm}\VAR{doctitle}}
\author{\VAR{name}}
\date{\today}

\usepackage{color}
\definecolor{deepblue}{rgb}{0,0,0.5}
\definecolor{deepred}{rgb}{0.6,0,0}
\definecolor{deepgreen}{rgb}{0,0.5,0}

\lstset{
    basicstyle=\ttfamily\small,
    breaklines=true,
    numbers=left,
    language=verilog,
    showstringspaces=false,
    keywordstyle=\color{deepblue},
    emphstyle=\color{deepred},    % Custom highlighting style
    stringstyle=\color{deepgreen},
}

\setlist[itemize]{noitemsep, topsep=5pt}
\setlist[enumerate]{noitemsep, topsep=5pt}

% for within multicol
\newenvironment{Figure}
  {\par\medskip\noindent\minipage{\linewidth}}
  {\endminipage\par\medskip}


\makeatletter
\pagestyle{fancy}
\fancyhf{}
\rhead{Page \thepage}
\lhead{\VAR{doctitle}}
\fancyhead[C]{\nouppercase\leftmark}
\makeatother

% supress section numbers
\renewcommand\sectionmark[1]{\markboth{#1}{}}

\begin{document}
\maketitle

% load with jinja
\VAR{content}
\end{document}
```

0.2 __init__.py

```
import os
import jinja2
from jinja2 import Template

# set up jinja
__location__ = os.path.realpath(
    os.path.join(os.getcwd(), os.path.dirname(__file__)))

print(__location__)

latex_jinja_env = jinja2.Environment(
        block_start_string = ’\BLOCK{’, block_end_string = ’}’,
        variable_start_string = ’\VAR{’,
        variable_end_string = ’}’,
        comment_start_string = ’\#{’,
        comment_end_string = ’}’,
        line_statement_prefix = ’%%’,
        line_comment_prefix = ’%#’,
        trim_blocks = True,
        autoescape = False,
        loader = jinja2.FileSystemLoader(__location__)
)

def generate(name, title):
    # make the directory to hold the writeup
    try:
        os.mkdir("code-writeup")
    except FileExistsError:
        confirm = input("Directory ’code-writeup’ exists. Delete it? (y) ")
        if confirm != ’y’:
            print("okay, exiting")
            quit()

    def get_filenames():
        for filename in os.listdir():
            if os.path.isfile(filename):
                yield filename

    tex_to_insert = ""
    for filename in get_filenames():
        with open(filename, "r") as f:
            file_contents = f.read()

        # make sure to escape underscores
        tex_to_insert += "\\subsection{\\tt{\\detokenize{" + filename + "}}}\n"
        tex_to_insert += "\\begin{lstlisting}\n"
        tex_to_insert += file_contents
        tex_to_insert += "\n\\end{lstlisting}\n"

    template = latex_jinja_env.get_template(’template.tex’)
    finished = template.render(name=name, doctitle=title, content=tex_to_insert)

    with open("code-writeup/writeup.tex", "w") as f:
        f.write(finished)

def main():
    # get user and document info
    name = input("What’s your name? ")
    title = input("What should this document be titled? ")

    generate(name, title)
```