```python
def hello_world():
    print("Hello floating world!")
```

Listing 1: Floating Listing.

This is Python code:

```python
1   import numpy as np
2
3   def incmatrix(genl1,genl2):
4       m = len(genl1)
5       n = len(genl2)
6       M = None #to become the incidence matrix
7       VT = np.zeros((n*m,1), int)  #dummy variable
8       print("I am here")
9
10      #compute the bitwise xor matrix
11      M1 = bitxormatrix(genl1)
12      M2 = np.triu(bitxormatrix(genl2),1)
13
14      for i in range(m-1):
15          for j in range(i+1, m):
16              [r,c] = np.where(M2 == M1[i,j])
17              for k in range(len(r)):
18                  VT[(i)*n + r[k]] = 1;
19                  VT[(i)*n + c[k]] = 1;
20                  VT[(j)*n + r[k]] = 1;
21                  VT[(j)*n + c[k]] = 1;
22
23                  if M is None:
24                      M = np.copy(VT)
25                  else:
26                      M = np.concatenate((M, VT), 1)
27
28                  VT = np.zeros((n*m,1), int)
29
30      return M
```

Here is an example with listing:
References: listing 1, 1
Here is file input:

```latex
% https://www.overleaf.com/learn/latex/Code_Highlighting_with_minted
% https://latex-tutorial.com/code-listings/
\documentclass{article}
```

```latex
\usepackage[outputdir=../out]{minted} % this is tricky, relative to current file.
\usemintedstyle[python]{monokai} % rather than for all language
\RequirePackage{xcolor}

\usepackage{cleveref}

\definecolor{bgcolor}{RGB}{16, 31, 62}

\begin{document}
    This is Python code:
% need -shell-escape
\begin{minted}[linenos, bgcolor=bgcolor]{python}
import numpy as np

def incmatrix(genl1,genl2):
    m = len(genl1)
    n = len(genl2)
    M = None #to become the incidence matrix
    VT = np.zeros((n*m,1), int)  #dummy variable
    print("I am here")

    #compute the bitwise xor matrix
    M1 = bitxormatrix(genl1)
    M2 = np.triu(bitxormatrix(genl2),1)

    for i in range(m-1):
        for j in range(i+1, m):
            [r,c] = np.where(M2 == M1[i,j])
            for k in range(len(r)):
                VT[(i)*n + r[k]] = 1;
                VT[(i)*n + c[k]] = 1;
                VT[(j)*n + r[k]] = 1;
                VT[(j)*n + c[k]] = 1;

                if M is None:
                    M = np.copy(VT)
                else:
                    M = np.concatenate((M, VT), 1)

                VT = np.zeros((n*m,1), int)

    return M
\end{minted}

    Here is an example with listing:
    \begin{listing}
```

```
    \begin{minted}{Python}
        def hello_world():
            print("Hello floating world!")
    \end{minted}
    \caption{Floating Listing.}
    \label{lst:hello}
\end{listing}

References: \cref{lst:hello},~\ref{lst:hello}

Here is file input:
\inputminted{tex}{source_code_test.tex}

Here is an example of a line code \mintinline{c}|int i| .
\end{document}
```

Here is an example of a line code `int i` .