# 473.https://stackoverflow.com/questions/70798513/why-is-solidity-vscode-intellisense-only-working-if-i-explicitly-specify-an-open

**T:**Why is Solidity Vscode Intellisense only working if i explicitly specify an Open Zeppelin contract

**Q:**I'm struggling severely to configure my Vscode working properly.  
  
I have imported contract of openzeppelin - ERC721.solNow I expected its function \_mint() be available in intellisense when i start typing it in my contract, but here is what i see if I start typing it and then click Ctrl+Space:  
  
  
  
In the same time if i explicitly type the contract name - then i can see intellisense autocompletion:  
  
  
  
How can i make it possible to simply get intellisense suggestion of this function once i start typing it?  
  
UPDATE: I provided a bigger code snippet and simplified to minimum for example purposes.  
  
UPDATE: I realised that the autocompletion now in the vscode solidity plugin works in a way that ignores the modules from node\_modules. Quick way to test it - I put my contract file into the openzeppelin module folder, next to the ERC721.sol - autocompletion works fine. (suggested by @marko-popovic)  
  
Question remains the same though: Isn't that possible to make it so that I could use autocompletion for modules that are present in node\_modules and imported into my contract?

**C1:**Could you show a bigger code snippet where this does not work, so that we can see how are you actually trying to use ERC21?

**C2:**Hey @MarkoPopovic I've provided a bigger code snippet

**C3:**Is the behavior the same for all methods of ERC721? For example, is there intellisense for methods name or transferFrom?

**C4:**Absolutely no methods, variables are not pulled up to intellisense, unless I explicitly specify the contract to import from. Even though the solidity vscode plugin claims to have this functionality github.com/juanfranblanco/vscode-solidity/blob/master/… This makes me think i'm doing something wrong

**C5:**Well in that video the files are local. In you case, ERC721 is imported from NPM (using @). It might be that the extension cannot figure it out in that case unless you give the ERC721.` prefix. You could test this easily. Create a local dummy contract in a separate file. Import the file, inherit from that contract and see if the intellisense works in that case.

1 **Answer**

**A1:**In you case, ERC721 is imported from NPM (using @), while the imported files in the extension demo video are in the same directory as the contract that is importing them.  
  
To enable intellisense, either keep using the NPM import style and prefix the method names with ERC721. (which would be my suggestion), or move the openzepellin code manually into the same folder as your contract.  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
Finally, you cannot achieve that with the current version of the extension, as it is obviously works properly, but has this limitation with different types of imports. This might be improved in some future version, but if you want to get that feature right now you will either have to move the code manually or maybe look for another extension.