# 306.https://stackoverflow.com/questions/71718798/definition-of-base-has-to-precede-definition-of-derived-contract-erc721-impleme

**T:**Definition of base has to precede definition of derived contract (ERC721 implementation)

**Q:**The top SO or ETH Stack Exchange answers don't seem to apply to my case (I could be wrong of course)  
  
I'm getting the error describer in the title in this file:  
  
// SPDX-License-Identifier: MITpragma solidity ^0.8.0;import "./ERC721Metadata.sol";import "./ERC721.sol";contract ERC721Connector is ERC721Metadata, ERC721 { // ^^^^^^^ (Definition of base has to precede definition of derived contract) constructor(string memory name, string memory symbol) ERC721Metadata(name, symbol) {}}  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
Here's what ERC721Metadadata looks like:  
  
// SPDX-License-Identifier: MITpragma solidity ^0.8.0;contract ERC721Metadata { string private \_name; string private \_symbol; constructor(string memory named, string memory symbolified) { \_name = named; \_symbol = symbolified; } function name() external view returns (string memory) { return \_name; } function symbol() external view returns (string memory) { return \_symbol; }}  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
And here is what ERC721 looks like:  
  
// SPDX-License-Identifier: MITpragma solidity ^0.8.0;import "./ERC721Metadata.sol";import "./ERC721Connector.sol";contract ERC721 { event Transfer( address indexed from, address indexed to, uint256 indexed tokenId ); mapping(uint256 => address) private \_tokenOwner; mapping(address => uint256) private \_OwnedTokensCount; function \_exists(uint256 tokenId) internal view returns (bool) { address owner = \_tokenOwner[tokenId]; return owner != address(0); } function \_mint(address to, uint256 tokenId) internal { require(to != address(0), "ERC721: minting to the zero address"); require( !\_exists(tokenId), "ERC721: minting a token that already exists (been minted)" ); \_tokenOwner[tokenId] = to; \_OwnedTokensCount[to] += 1; emit Transfer(address(0), to, tokenId); }}  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]

1 **Answer**

**A1:**Do you need to import ERC721Connector contract in your ERC721 contract? If not, you can remove  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
import "./ERC721Connector.sol"; // line 4, ERC721.sol  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
from your file and it should work fine. Your imports are causing the issue,  
  
ERC721Connector tries to import ERC721, but ERC721 also needs ERC721Connector so the compiler says  
  
WARN: THIS PARAGRAPH CONTAINS TAG: [CODE]   
  
Definition of base has to precede definition of derived contract  
  
Because the base contract doesn't precede the defined contract.