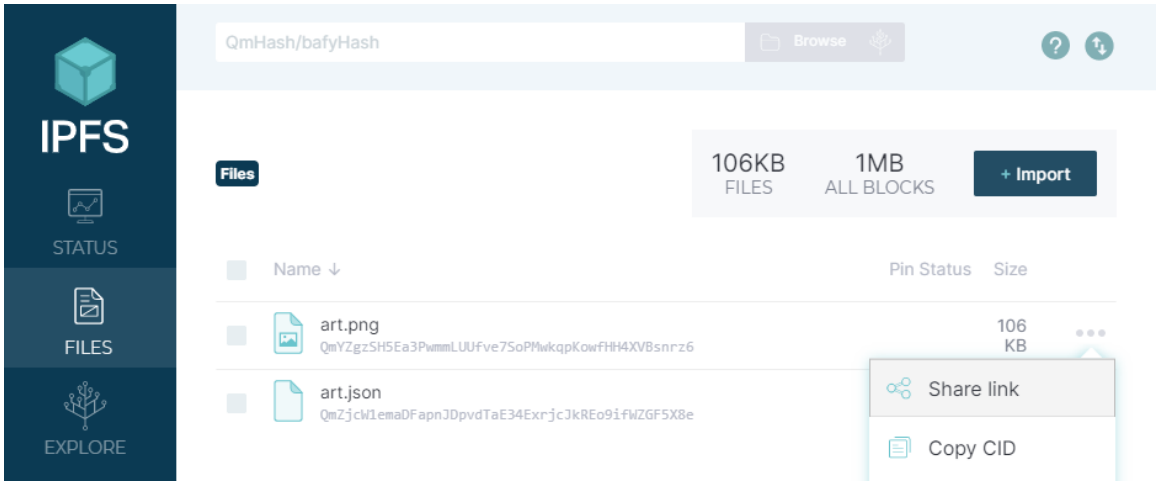
Simple NFT

Download the IPFS-Desktop at: https://github.com/ipfs/ipfs-desktop

Prepare your IPFS item to be linked to the NFT token.

1. Select an image as your NFT. Open the IPFS Desktop, select Files, and drag the image into it.



1. Create a json metadata to represent your NFT. *Copy the CID* (content identifier) of the image file (in example below: art.png) which is the hash: QmYZgzSH5Ea3PwmmLUUfve7SoPMwkqpKowfHH4XVBsnrz6
2. Add in the image CID in the image field with https://ipfs.io/ipfs/ in front. So that the json metadata links to the image file.

```
{
    "name": "NFT Art",
    "description": "This image shows the true nature of NFT as art",
    "image": "https://ipfs.io/ipfs/QmYZgzSH5Ea3PwmmLUUfve7SoPMwkqpKowfHH4XVBsnrz6",
}
```

3. Drag art.json into IPFS as well

Completed will look like the image above.

Compile NFT Contract

Now you are ready to create the NFT Contract

Run this code in Remix with Injected Web3 and Metamask (make sure you have some testnet ETH, Kovan if possible).

// SPDX-License-Identifier: MIT
pragma solidity 0.8.0;

```
import "https://github.com/0xcert/ethereum-erc721/src/contracts/tokens/nf-token-metadata.sol";
import "https://github.com/0xcert/ethereum-erc721/src/contracts/ownership/ownable.sol";

contract newNFT is NFTokenMetadata, Ownable {

 constructor() {
  nftName = "Synth NFT";
  nftSymbol = "SYN";
 }

 function mint(address _to, uint256 _tokenId, string calldata _uri) external onlyOwner {
  super._mint(_to, _tokenId);
  super._setTokenUri(_tokenId, _uri);
 }

}
```

Import code is at:
https://github.com/0xcert/ethereum-erc721/blob/master/src/contracts/tokens/nf-token.sol

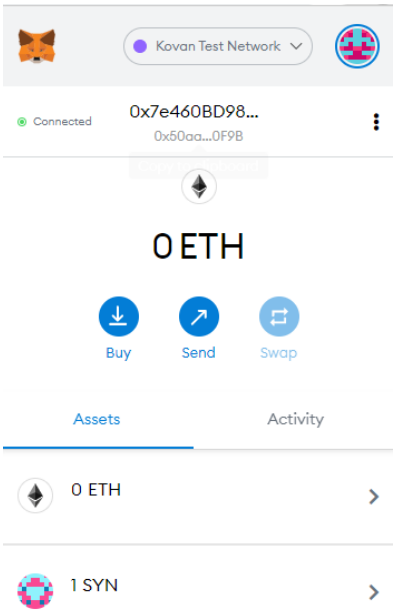
Deploy nft.sol

Go ahead and deploy nft.sol (check that it is not other contract).

Mint Token

1. Click on the ***mint*** button. Fill the **_to** address with the owner of the token (any address you choose).
2. Give a unique token id to **_tokenId**, say 1
3. In **_uri** put in the full address of the json file that you have created earlier. In the example:
   https://ipfs.io/ipfs/QmZjcW1emaDFapnJDpvdTaE34ExrjcJkREo9ifWZGF5X8e?filename=art.json
   You can this URI in the Share Link menu.
4. You can check that the owner of the token id with ***balanceOf*** and ***ownerOf***
5. ***tokenURI*** gives the uri for the token id.

Shared Contract
Deploy this contract newNFT.sol (my contract is at 0x0782c05Fd77941Aa33cf1A323eE206afCa9F0247) in Kovan TestNet so that we all run the same contract. Now we can mint (only owner) and transfer (using approve and safeTransferFrom) to each other.



Task

Currently the contract allows the same tokenURI to minted to different tokenID. How would you prevent that from happening? Can you make the changes in code?