

Dosya indirme Linki: <https://github.com/furkankocak485/ERC-721-LowGasFee-Smart-Contract/archive/refs/heads/main.zip> - https://github.com/HashLips/hashlips_nft_contract/archive/refs/tags/v1.0.1.zip
 Youtube Video Kaynağı: <https://www.youtube.com/watch?v=McmhpmnQLto> - <https://www.youtube.com/watch?v=gGBupa19mh0&t=1523s>

FURKAN KOÇAK

İlerleyiş Sırası: Sözleşme Düzenleme - Deploy - Verify - Mint - setPaused (true) - Listeleme

[Remix.Ethereum]

Sözleşmedeki Düzenleme Kısımları

```
// Koleksiyon adı
contract NFTCOLLECTION is ERC721, Ownable {
  using Strings for uint256;
  using Counters for Counters.Counter;

  Counters.Counter private supply;
  // NFT Görsellerinin ana metadata CID kodu (uriPrefix)
  string public uriPrefix = "ipfs://__CID-Metadata__";
  string public uriSuffix = ".json";
  string public hiddenMetadataUri;

  uint256 public cost = 0.00001 ether; // NFT Koleksiyon Eth Değeri
  uint256 public maxSupply = 250; // NFT Koleksiyon Parça Miktarı
  uint256 public maxMintAmountPerTx = 50; // NFT Koleksiyonun Cüzdan Başlı Mint Miktarı

  bool public paused = false; // NFT Sözleşme Mintlemeye Açık (false), Mintlemeye kapalı (true)
  bool public revealed = true; // NFT Sözleşmedeki parçalar görünür (true), parçalar görünürlüğü kapalı (false)

  // NFT: NAME-SYMBOL YAZILACAK.
  constructor() ERC721("NAME", "SYMBOL") {
    setHiddenMetadataUri("ipfs://__HiddenCIDMetadata__/hidden.png"); // NFT GİZLİ (hidden.json) METADATA
  }

  // NFT Sözleşmeden Para Çekme (sahiplik bölümü)
  function withdraw() public onlyOwner {
    // ===== |PAY VERİLECEKSE BU KISIM DURACAK|
    // Bu koleksiyon için pay vereceğin cüzdan (Bu kısımda verilecek payın yüzdelik kısmı)
    (bool hs, ) = payable(0x2EB46561F11F1f2DaEC1D9d256149022A3a53).call{value: address(this).balance * 5 / 100}("");
    require(hs);
    // ===== |PAY VERİLMEYECEKSE ÜST KISIM SİL|
    (bool os, ) = payable(owner()).call{value: address(this).balance}("");
    require(os);
  }

  // =====
  function _mintLoop(address _receiver, uint256 _mintAmount) internal {
```

[SÖZLEŞME DÜZENLEME]

- <https://remix.ethereum.org/> adresine git.
- Workspace kısmından [+] ile yeni bir çalışma alanı oluştur. Contract klasörü içerisine yeni dosya ekle.
- Erc721- NFT Smart-Contract.sol içerisindeki kodları kopyala. Contract içindeki yeni dosya içine yapıştır. Sözleşme üzerindeki düzenlemeleri yap. SOLIDITY COMPILER yeşil tik gelmişse her şey yolunda.
- SOLIDITY COMPILER kısmından Compiler kısmının pragma aralığında olduğundan emin ol. Genelde 0.8.7 olur. Solidity - compiler default - Auto compile - Enable optimization 200'ü seçiyoruz.

[DEPLOY]

- DEPLOY & RUN TRANSACTIONS bölümünden Injected Web3 seçiyoruz ve cüzdanımızın doğruluğunu ACCOUNT bölümünden kontrol ediyoruz. Ağları öğrenmek için <https://chainlist.org/> adresine gitmek yeterlidir.
- DEPLOY & RUN TRANSACTIONS/Contract bölümünden sözleşmemizi seçiyoruz.
- Deploy tuşuna basıyoruz. Metamask'ımızdan gerekli imza onayını verip sözleşmeyi dağıtıyoruz.
- Dağıtılan sözleşmenin adresini kopyala ve bir yere kaydet.
- Sözleşmeyi çağırmak için At Address bölümünü kullan. Turuncu kısım yazma, mavi kısım okuma kısmıdır.

[VERIFY]

- PLUGIN MANAGER üzerinden flattener yazıp arat. FLATTENER eklentisini Activate et.
- FLATTENER kısmından Flatten contracts/sözleşmeadi.sol a tıkla. Sonrasında Saves sözleşmeadi.sol tıkla. Çıkan pencereden Remember this choice'ı tik at. Son olarakta Accept'e bas. Sözleşme ana dosyasına direk doğrulanmış sözleşme .sol formatında çıkacaktır. İster Doğrulanmış dosyayı düzenle ister direkt kullan.
- Etherscan üzerinden sözleşme kodunu yapıştır arat. contract kısmından verify and publish'e bas.
- Sırasıyla Contract Address (Sözleşme Adresi) - Compiler Type (Solidity (Single File)) - Compiler Version (Deploy öncesi kullanılan compiler sürümü) - License Type (3) MIT License (MIT)) seç.
- Çıkan sayfadaki bölümleri doldur. Contract Address: Sözleşme Adresimiz. Optimization'u Yes yapıyoruz. Compiler: Sözleşmeyi deploy ederken kullandığımız sürümü seçiyoruz. Enter the Solidity Contract Code below: Bu kısma FLATTENER ile çıkarttığımız sözleşmenin yazısını olduğu gibi ya da düzenlenmiş yani (Açıklamaları silinmiş) halini yapıştırıyoruz. Sonrasında I'm not a robot'u doldurup, Verify and Publish'e basıyoruz. Böylelikle sözleşmemiz onaylanmış ve etherscan üzerinden yazılabilir, okunabilir ve görülebilir hale geliyor.

Dosya indirme Linki: <https://github.com/furkankocak485/ERC-721-LowGasFee-Smart-Contract/archive/refs/heads/main.zip> - https://github.com/HashLips/hashlips_nft_contract/archive/refs/tags/v1.0.1.zip
 Youtube Video Kaynağı: <https://www.youtube.com/watch?v=McmhpmnQLto> - <https://www.youtube.com/watch?v=gGBupa19mh0&t=1523s>

FURKAN KOÇAK

[MINT]

- Mint kısmına kaç adet basacağını yaz. Cost kısmındaki Wei formatındaki NFT mint fiyatının biraz fazlasını **VALUE** kısmına yapıştır. (Bu wei türünden ödeyeceğimiz gasfee ücretimiz olacak.) Mente bas.

[setPaused – SÖZLEŞME BASIMINI DURDURMA]

- setPaused kısmına true yaz ve butona bas. Sözleşme basımını durdurur yani nftmintlenemez hale getirir.

[LİSTELEME]

- Hangi aşda dağıtım yapılmışsa o ağı kullanan marketlerde listeleme yaparak NFTlerini satışa sun.

[YAZMA BÖLÜMÜ]

[OKUMA BÖLÜMÜ]

At Address	Akıllı sözleşme adresini buraya yapıştır ve sözleşme komutlarını geri çağır.
approve	address to, uint256 tokenId
mint	Koleksiyon Cüzdanına basmak için sayı girilip mint'e basılır.
mintForAddress	Bir cüzdana nft basmak için kullanılır.
renounceOwn...	
safeTransferFr...	Başka bir cüzdana nft transferi için kullanılır.
safeTransferFr...	address from, address to, uint256 tokenId, bytes _data
setApprovalFo...	address operator, bool approved
setCost	NFT fiyatlandırmasını güncellemek için kullanılır. (Wei formatında)
setHiddenMet...	Hidden (gizli nft) metadata adresi. (hidden.json dosyası)
setMaxMintA...	Cüzdana başı basma sayısını güncellemek için kullanılır.
setPaused	NFT Sözleşme basmaya Açık (false), basmaya kapalı (true).
setRevealed	NFT Sözleşmedeki parçalar görünür (true), parçalar görünürlüğü kapalı (false)
setUriPrefix	NFT Görsellerinin Ana CID Metadata dosyası. "ipfs://__CID-Metadata__/"
setUriSuffix	.json dosyaları
transferFrom	İstenilen bir NFT (TokenID) cüzdandan cüzdana aktarma.
transferOwner...	address newOwner
withdraw	Kazanılan ethereum gelirini cüzdanımıza çekmeye yarar.

balanceOf	address owner
cost	Wei cinsinden bir parçanın fiyatı buradan öğrenilir. NFT mintlerken lazım olacak
getApproved	uint256 tokenId
hiddenMetada...	hidden (GİZLİ NFT) CID metadata ipfs URI'si
isApprovedFor...	address owner, address operator
maxMintAmou...	Cüzdan başı kaç nft basabileceği tutarı gösterir.
maxSupply	Toplam nft miktarı
name	Sözleşme Adı
owner	Sözleşmenin dağıtıldığı cüzdan adresi / NFTlerin Sahibi (Creator)
ownerOf	uint256 tokenId
paused	NFT Sözleşme Mintlemeye Açık (false), Mintlemeye kapalı (true)
revealed	NFT Sözleşmedeki parçalar görünür (true) parçalar görünürlüğü kapalı (false)
supportsInterf...	bytes4 interfaced
symbol	NFT sözleşmesinin Symbolü
tokenURI	NFT URI'lerini çağırır.
totalSupply	Kaç nft basıldığını gösterir.
uriPrefix	NFT Görsellerinin ana metadata CID kodu. "ipfs://__CID-Metadata__/"
uriSuffix	NFT Görsellerinin ana metadata CID dosya türünü gösterir.
walletOfOwner	Bir cüzdanın hangi nftlere sahip olduğunu gösterir.

FURKAN KOÇAK