Questions	
Write a smart contract in Solidity that implements a basic token.	
Write a function that takes in a block header and verifies its validity.	
Implement a Merkle tree data structure and use it to validate a set of transactions.	
Write a script that extracts data from a blockchain and stores it in a database.	
Develop a DApp that interacts with a smart contract on the Ethereum network.	
Write a program that calculates the average gas price for transactions on the Ethereum network over a specific time period.	
Implement a consensus algorithm such as Proof of Work or Proof of Stake.	
Develop a custom blockchain and create a client to interact with it.	
Write a program that generates a random private key for a Bitcoin wallet.	
Implement a secure hashing algorithm like SHA-256 or Keccak.	
Implement a distributed file system that uses blockchain to store the file hashes and verify the integrity of the files.	
Develop a privacy-focused blockchain that uses zero-knowledge proofs to protect the user's data.	
Write a program that simulates a blockchain network with multiple nodes and implements a consensus algorithm like Proof of Work or Proof of Stake.	
Implement a decentralized exchange on the Ethereum network that allows for trading of ERC-20 tokens.	
Develop a smart contract that implements a decentralized auction with sealed bids.	
Create a program that generates a hierarchical deterministic wallet using BIP-39 and BIP-44 standards.	
Write a program that generates a Bitcoin address from a given public key.	
Implement a blockchain-based voting system that allows for secure and transparent voting.	
Develop a DApp that uses IPFS for file storage and retrieval.	
Implement a payment channel on the Lightning Network for fast and low-cost Bitcoin transactions.	
Develop a decentralized identity system that allows users to own and control their own identity data.	
Implement a sidechain that is interoperable with the Ethereum network.	
Write a program that uses a zero-knowledge proof to verify the correctness of a computation.	
Create a smart contract that implements a dynamic pricing model for a decentralized storage system.	
Develop a DApp that uses a decentralized oracle to retrieve data from external sources.	
Write a program that generates a multi-signature Bitcoin transaction.	
Implement a consensus algorithm that uses sharding to increase the scalability of the blockchain.	
Develop a decentralized messaging platform that uses encryption to ensure message privacy.	
Write a smart contract that implements a decentralized crowdfunding platform.	
Create a program that monitors the blockchain for specific events and triggers actions based on those events.	
Develop a decentralized reputation system that uses smart contracts to track and verify reputation scores.	
Write a program that implements a proof-of-authority consensus algorithm.	
Implement a cross-chain atomic swap that allows for the exchange of cryptocurrencies across different blockchains.	
Create a DApp that uses an off-chain scaling solution like Plasma or Raiden to improve the scalability of the application.	
Develop a smart contract that implements a decentralized prediction market.	
Write a program that uses a homomorphic encryption scheme to perform secure computations on encrypted data.	
Implement a blockchain-based supply chain management system that uses smart contracts to track the movement of goods.	
Develop a decentralized identity verification system that uses a zero-knowledge proof to verify user data.	
Write a program that implements a ring signature scheme for anonymous transactions.	
Implement a blockchain-based game that uses non-fungible tokens (NFTs) to represent unique in-game assets.	
Develop a privacy-preserving decentralized exchange that uses ring signatures and stealth addresses to protect user privacy.	
Write a program that implements a Byzantine fault-tolerant consensus algorithm to ensure the integrity and security of the blockchain.	
Implement a blockchain-based insurance platform that uses smart contracts to automate the claims process.	
Create a DApp that uses a sidechain to improve the scalability of the application and reduce gas costs.	
Develop a smart contract that implements a decentralized token issuance and distribution platform.	
Write a program that uses a hash function to generate a unique identifier for a piece of data that can be used to verify its authenticity.	
Implement a blockchain-based system for tracking carbon credits and other environmental assets.	
Develop a DApp that uses IPFS and the InterPlanetary Linked Data (IPLD) protocol to store and retrieve data in a decentralized manner.	
Write a program that implements a zero-knowledge proof scheme for secure and private data exchange.	
Create a smart contract that implements a decentralized lottery with verifiable randomness.	
Develop a decentralized marketplace that uses smart contracts to handle transactions and escrow.	
Write a program that implements a directed acyclic graph (DAG) consensus algorithm for a blockchain.	
Implement a blockchain-based voting system that uses smart contracts to ensure the integrity and transparency of the voting process.	
Create a DApp that uses a sidechain with a unique governance mechanism to improve scalability and security.	
Develop a smart contract that implements a decentralized affiliate marketing platform.	
Write a program that uses a Schnorr signature scheme to sign transactions and verify their authenticity.	
Implement a blockchain-based real estate platform that uses smart contracts to automate the buying and selling process.	
Develop a DApp that uses a decentralized file storage system like Filecoin or Storj to store and retrieve data.	
Write a program that implements a secure multiparty computation protocol for private data sharing.	
Create a smart contract that implements a decentralized exchange for non-fungible tokens (NFTs).	
Develop a smart contract that implements a decentralized lending and borrowing platform.	
Write a program that implements a proof-of-stake consensus algorithm to validate transactions on the blockchain.	
Implement a blockchain-based reputation system that uses token incentives to reward positive behavior and punish bad actors.	
Create a DApp that uses a privacy-preserving protocol like Aztec or Zether to enable private transactions on the blockchain.	
Develop a smart contract that implements a decentralized crowdfunding platform.	
Write a program that uses a polynomial commitment scheme to verify the integrity of data stored on the blockchain.	
Implement a blockchain-based healthcare platform that uses smart contracts to manage patient data and ensure privacy.	
Develop a DApp that uses a decentralized oracle to provide off-chain data to the blockchain.	
Write a program that implements a state channel protocol to enable fast and cheap off-chain transactions.	
Create a smart contract that implements a decentralized escrow service for peer-to-peer transactions.	
Develop a smart contract that implements a decentralized prediction market using a scalar voting system.	
Write a program that implements a consensus protocol based on gossip and rumor spreading in a blockchain network.	
Implement a blockchain-based supply chain management system that uses smart contracts to track goods and ensure transparency.	
Create a DApp that uses a zero-knowledge proof system like ZK-SNARKs to enable anonymous transactions on the blockchain.	
Develop a smart contract that implements a decentralized insurance platform.	
Write a program that uses a homomorphic encryption scheme to protect sensitive data on the blockchain.	
Implement a blockchain-based identity verification system that uses biometric data to authenticate users.	
Develop a DApp that uses a decentralized reputation system to rate and review products or services.	
Create a smart contract that implements a decentralized autonomous organization (DAO).	
Write a program that uses a threshold signature scheme to enable multisig transactions on the blockchain.	
Develop a smart contract that implements a decentralized autonomous marketplace using a multi-token model.	
Write a program that implements a consensus protocol based on a Byzantine fault-tolerant algorithm for a blockchain network.	
Implement a blockchain-based carbon credits trading platform that uses smart contracts to incentivize sustainable practices.	

Create a DApp that uses a sharding system to improve the scalability of the blockchain.

Develop a smart contract that implements a decentralized real-time bidding advertising platform.

Write a program that uses a threshold cryptosystem to enable distributed key generation and signature verification on the blockchain.

Implement a blockchain-based gaming platform that uses non-fungible tokens (NFTs) to represent in-game assets.

Develop a DApp that uses a decentralized data storage system like IPFS to store and retrieve data.

Create a smart contract that implements a decentralized reputation system for freelancers and independent contractors.

Write a program that uses a distributed randomness protocol to generate unpredictable values for use in blockchain applications.

Develop a smart contract that implements a decentralized asset management platform using a multi-collateral model.

Write a program that implements a consensus protocol based on a proof-of-stake algorithm for a blockchain network.

Implement a blockchain-based voting platform that uses smart contracts to ensure the integrity of the voting process.

Create a DApp that uses a cross-chain protocol to enable interoperability between different blockchain networks.

Develop a smart contract that implements a decentralized exchange using a liquidity pooling model.

Write a program that uses a multi-party computation protocol to enable secure and private computation on the blockchain.

Implement a blockchain-based real estate platform that uses smart contracts to manage property ownership and transfers.

Develop a DApp that uses a decentralized oracle system to retrieve off-chain data for use on the blockchain.

Create a smart contract that implements a decentralized crowdfunding platform using a milestone-based model.

Write a program that uses a distributed file system like Sia to enable decentralized storage of large files on the blockchain.

Solidity Questions

Write a Solidity function to check if a given address is a contract or not.

Write a Solidity function to transfer tokens from one address to another.

Write a Solidity function to withdraw funds from a smart contract.

Write a Solidity function to check the balance of a given address.

Write a Solidity function to implement a time-locked contract, which allows funds to be withdrawn only after a certain time has elapsed.

Write a Solidity function to implement a voting system, where each address can vote only once.

Write a Solidity function to implement a basic ERC-20 token.

Write a Solidity function to implement a crowdsale, where tokens are sold in exchange for ether.

Write a Solidity function to implement a decentralized exchange, where users can trade ERC-20 tokens.

Write a Solidity function to implement a multi-signature wallet, where funds can be released only with the approval of multiple addresses.

Write a Solidity function to implement a staking system, where users can earn rewards for holding tokens.

Write a Solidity function to implement a lottery, where users can buy tickets for a chance to win a prize.

Write a Solidity function to implement a blind auction, where bidders submit sealed bids and the highest bidder wins.

Write a Solidity function to implement a rentable storage system, where users can rent storage space in exchange for tokens.

Write a Solidity function to implement a trustless escrow system, where funds are held in escrow until certain conditions are met.

Write a Solidity function to implement a decentralized identity system, where users can prove their identity without relying on a centralized authority.

Write a Solidity function to implement a prediction market, where users can bet on the outcome of future events.

Write a Solidity function to implement a supply chain management system, where products can be tracked from creation to delivery.

Write a Solidity function to implement a decentralized autonomous organization (DAO), where users can vote on governance decisions.

Write a Solidity function to implement a smart contract insurance policy, where users can be compensated for losses that meet certain conditions.

Write a Solidity function to implement a token swap, where one ERC-20 token can be exchanged for another.

Write a Solidity function to implement a token vesting contract, where tokens are gradually released over a period of time.

Write a Solidity function to implement a gasless transfer, where tokens can be transferred without requiring the user to pay for gas.

Write a Solidity function to implement a cross-chain bridge, where tokens can be transferred between different blockchains.

Write a Solidity function to implement a decentralized file storage system, where users can store and retrieve files without relying on a centralized server.

Write a Solidity function to implement a non-fungible token (NFT) contract, where each token is unique and can represent a digital asset such as artwork or collectibles.

Write a Solidity function to implement a tokenized real estate contract, where users can invest in a property and receive a proportional share of the rental income.

Write a Solidity function to implement a decentralized identity verification system, where users can prove their identity using a network of trusted validators.

Write a Solidity function to implement a decentralized marketplace, where users can buy and sell goods and services without relying on a centralized platform.

Write a Solidity function to implement a stablecoin contract, where the value of the token is pegged to a stable asset such as the US dollar.

Write a Solidity function to implement a prediction market for sports events, where users can bet on the outcome of games and matches.

Write a Solidity function to implement a lending platform, where users can borrow and lend tokens with interest.

Write a Solidity function to implement a decentralized exchange (DEX) that uses an automated market maker (AMM) system.

Write a Solidity function to implement a smart contract lottery, where the winner is chosen randomly and transparently using a verifiable random function (VRF).

Write a Solidity function to implement a decentralized autonomous liquidity pool (DALP), where users can pool their liquidity and earn rewards in exchange for providing liquidity to the pool.

Write a Solidity function to implement a yield farming contract, where users can earn rewards by staking tokens in a liquidity pool.

Write a Solidity function to implement a decentralized identity system for healthcare, where users can share their medical records with trusted parties.

Write a Solidity function to implement a decentralized governance platform, where users can vote on proposals and make decisions about the direction of a project.

Write a Solidity function to implement a decentralized token swap aggregator, where users can find the best price for a token across multiple DEXs.

Write a Solidity function to implement a decentralized social media platform, where users can post and interact with content without relying on a centralized authority.

Write a Solidity function to implement a decentralized reputation system, where users can earn reputation points based on their contributions to a project.

Write a Solidity function to implement a decentralized insurance platform, where users can purchase insurance policies and file claims without relying on a centralized insurer.

Write a Solidity function to implement a decentralized job marketplace, where users can find work and hire freelancers without relying on a centralized platform.

Write a Solidity function to implement a decentralized advertising platform, where users can earn rewards for viewing and interacting with ads.

Write a Solidity function to implement a decentralized real-time bidding platform, where advertisers can bid on ad space in real time.

Write a Solidity function to implement a decentralized credit scoring system, where users can receive credit scores based on their financial behavior and use these scores to obtain loans.

Write a Solidity function to implement a decentralized voting system, where users can cast votes that are counted in a transparent and tamper-proof way.

Write a Solidity function to implement a decentralized data marketplace, where users can buy and sell data without relying on a centralized authority.

Write a Solidity function to implement a decentralized energy trading platform, where users can buy and sell renewable energy certificates (RECs).

Write a Solidity function to implement a decentralized prediction market for weather events, where users can bet on the outcome of weather events such as hurricanes and tornadoes.

Write a Solidity function to implement a decentralized music streaming platform, where users can stream music and earn rewards for creating and sharing playlists.

Write a Solidity function to implement a decentralized peer-to-peer lending platform, where users can borrow and lend money without relying on a centralized lender.

Write a Solidity function to implement a decentralized ride-sharing platform, where users can find and offer rides without relying on a centralized platform.

Write a Solidity function to implement a decentralized supply chain financing platform, where suppliers can obtain financing based on their receivables.

Write a Solidity function to implement a decentralized sports betting platform, where users can bet on the outcome of sports events such as football and basketball games.