PassQueen
a.k.a dumpsqueen.com

Certified Blockchain Business Foundations

Blockchain CBBF

Version Demo

Total Demo Questions: 10

Total Premium Questions: 107

Buy Premium PDF

https://passqueen.com

support@passqueen.com

QUESTION NO: 1

Which of the following are practical use cases for Blockchain?

A. Voting

B. A video sharing website

C. Medical records

D. Identity management

E. All of the above

ANSWER: E

Explanation:

All of the above is correct because blockchain can support each listed use case when a shared, tamper-evident record and coordination among multiple parties provide value. In voting, a blockchain-based system can provide an auditable record of ballot-related events, subject to rigorous requirements for voter privacy, eligibility, security, and accessibility. For medical records, blockchain can help establish an immutable audit trail for record access, consent, and data-sharing events; sensitive clinical data is generally kept off-chain and linked through secure references. Identity management can use verifiable credentials and decentralized identifiers to enable individuals or organizations to prove specific claims while retaining greater control over credential sharing. Video-sharing services can also use blockchain components for creator attribution, content rights, payment distribution, tokenized incentives, or decentralized storage and delivery architectures. These are practical application areas rather than a guarantee that blockchain is the best technical choice in every implementation. The appropriate design depends on governance, privacy, scale, legal obligations, and whether participants need a shared source of truth without relying on a single operator. NIST describes blockchain as a tamper-evident, tamper-resistant distributed ledger technology, a property that underpins these potential applications. See NISTIR 8202: Blockchain Technology Overview and the W3C Decentralized Identifiers specification.

QUESTION NO: 2

What can occur if one entity controls a majority of a Proof-of-Work network's hashing power? Select all that apply.

A. Reorganize recent blocks by producing a chain with more cumulative work

B. Attempt to double spend the attacker's own recent transactions

C. Delay or censor selected transactions

D. Spend another user's funds without that user's private key

E. Create unlimited coins outside the protocol rules

ANSWER: A B C

Explanation:

An entity with majority hashing power may be able to create an alternative chain with greater cumulative work and use it to reorganize recent transaction history. This capability can enable double spending of the attacker's own recent transactions and can allow the attacker to delay or censor selected transactions by refusing to include them in blocks.

Majority hashing power does not give the attacker the ability to create coins outside the protocol rules, spend coins belonging to other users without their private keys, or change consensus rules merely by operating mining hardware. Full nodes continue to validate blocks against the protocol's rules. Bitcoin's original paper describes the risk that a majority of CPU power could outpace honest nodes and revise recent history. See the Bitcoin white paper and the Bitcoin Developer Guide on the block chain.

QUESTION NO: 3

One of the big issues that new financial transaction Blockchain solutions face is_____.

A. increasing the number of transactions per second (TPS)

B. the inability to find use cases

C. solving user identity

D. getting banks to cooperate

ANSWER: A

Explanation:

Increasing the number of transactions per second (TPS) is correct because transaction throughput, often discussed as scalability, is a central requirement for blockchain systems intended to support financial activity at meaningful scale. A financial-transaction network must be able to validate, order, and settle a large volume of transfers reliably, including during periods of peak demand. In many blockchain designs, throughput is constrained by the time required to propagate transactions and blocks across distributed nodes and to reach consensus while maintaining security and decentralization. This creates a practical design challenge: raising capacity without weakening the integrity, availability, or trust assumptions that make a blockchain useful.

For financial services, the issue is especially important because payment, trading, clearing, and settlement workflows can generate high transaction volumes and may require predictable processing times. Blockchain platforms therefore address TPS through approaches such as protocol improvements, layer-two networks, batching, and other scaling architectures. Ethereum's documentation describes scalability as the ability to handle increased demand and identifies layer-two solutions as a key approach to increasing transaction throughput: https://ethereum.org/en/developers/docs/scaling/. The Bank for International Settlements also discusses the operational and scalability considerations relevant to distributed ledger technology in financial markets: https://www.bis.org/publ/qtrpdf/r_qt1709f.htm.

QUESTION NO: 4

Which risks should an organization consider before deploying a smart contract? Select all that apply.

A. Errors in contract logic

B. Compromise of an authorized private key

C. The impossibility of performing any code review

D. Automatic legal enforceability in every jurisdiction

ANSWER: A B

Explanation:

Smart-contract deployment requires careful review because code may control valuable assets or enforce business rules automatically after deployment. Errors in contract logic can lead to unintended transfers, inaccessible assets, or incorrect state changes. Private-key compromise can allow an attacker to submit transactions using the authority associated with the compromised key. The organization must also consider oracle reliability where the contract depends on off-chain data, as well as upgrade procedures, access controls, testing, audit practices, and incident response. The use of a blockchain does not guarantee that deployed code is free from vulnerabilities or that the business process is appropriate. Ethereum recommends security considerations and testing practices for smart-contract developers. See the Ethereum smart contract security documentation.

QUESTION NO: 5

What are benefits of using known identities in a permissioned Blockchain network? Select all that apply.

A. Accountability for participant actions

B. Enforcement of role-based access policies

C. Support for compliance and audit requirements

D. Automatic public visibility of all business data

E. Elimination of the need for transaction validation

ANSWER: A B C

Explanation:

Known identities support accountability because network participants can be associated with authenticated organizations or individuals. They also allow the network to enforce policies that define which participants may submit transactions, endorse proposals, operate nodes, or access selected resources. These capabilities are especially important in business networks that have contractual, regulatory, or audit requirements.

Known identity does not make data public to all participants, nor does it eliminate the need for consensus. Permissioned networks can use identity and access controls while retaining distributed validation and shared ledger processes. Hyperledger Fabric uses certificates and membership service providers to establish trusted identities and apply policies within a network. See the Hyperledger Fabric identity documentation and the Fabric membership documentation.

QUESTION NO: 6

Which statements describe tokenization? Select all that apply.

A. It can represent an asset or right as a digital token

B. It can apply programmed transfer rules to a digital asset

C. It automatically establishes legal ownership in every jurisdiction

D. It requires all tokenized assets to be publicly visible

ANSWER: A B

Explanation:

Tokenization is the representation of an asset, right, or unit of value as a digital token on a blockchain or other distributed ledger. A token can represent a native cryptoasset, a claim on an off-chain asset, a membership right, a loyalty point, or another defined interest. Tokenization can support more efficient transfer, automated rules, and improved traceability, depending on the design. However, recording a token on a blockchain does not by itself establish legal ownership of the underlying asset or remove the need for custody, compliance, and contractual arrangements. The relationship between the token and any off-chain asset must be clearly governed and enforceable. See the SEC information on crypto assets for regulatory considerations involving crypto assets.

QUESTION NO: 7

Why is a private Blockchain used instead of a public one?

A. To limit user access and permissions

B. It is less expensive to develop

C. It requires less mining power than a public Blockchain

D. It requires fewer administrators to manage it

ANSWER: A

Explanation:

To limit user access and permissions is correct because a private blockchain is designed for a defined group of known, authorized participants rather than unrestricted public participation. The organization or consortium operating the network can control who may join, submit transactions, operate validating nodes, and view particular ledger data. This permissioning supports business use cases in which participants need to share trusted records while maintaining governance, confidentiality, role-based responsibilities, and compliance with contractual or regulatory requirements.

In an enterprise setting, identity is therefore a core feature of the network model: participants are admitted under rules set by the network's governing authority, and permissions can be tailored to their roles. This does not remove the value of a blockchain ledger; it applies distributed-ledger capabilities within a controlled membership environment. IBM describes private blockchains as permissioned networks in which access is restricted to invited participants. Hyperledger Fabric similarly uses a permissioned approach and provides mechanisms for identity, membership, and access control. These characteristics make limiting access and permissions the fundamental reason to choose a private blockchain. See IBM's overview of private blockchains and the Hyperledger Fabric identity documentation.

QUESTION NO: 8

Which characteristics are commonly associated with smart contracts? Select all that apply.

A. They execute programmed rules on a blockchain

B. They can automate defined business actions

C. They independently verify the truth of all external data

D. They guarantee that an application is legally compliant

ANSWER: A B

Explanation:

Smart contracts are programs deployed to a blockchain that execute according to coded conditions. Their execution and resulting state changes can be validated by the network, which makes the contract logic and its outcomes auditable according to the rules of the relevant blockchain. Smart contracts can automate actions such as transferring tokens, recording approvals, or enforcing workflow steps when defined conditions are satisfied. However, a smart contract cannot inherently determine whether off-chain information is true. When it needs external data, such as a market price or delivery status, it must rely on an oracle or another trusted data-input mechanism. Smart contracts also do not eliminate the need for sound legal, security, governance, and operational design. See the Ethereum documentation on smart contracts.

QUESTION NO: 9

Blockchain transaction code categorizes the data put into blocks.

A. True

B. False

ANSWER: B

Explanation:

False is correct because a blockchain's data organization is defined primarily by its protocol's transaction and block data structures, not by a general concept called "transaction code" that categorizes data placed into blocks. A transaction is a structured record that expresses a state change or transfer according to the rules of a particular blockchain. Valid transactions are propagated, validated, and then included in blocks by block producers. Each block normally contains a header and an ordered collection of transaction records; the protocol specifies the fields, encoding, validation rules, and the way the transactions are committed to the block.

For example, Bitcoin documentation describes transactions as the fundamental data structures that transfer value and explains that blocks aggregate transactions into the ledger. Ethereum likewise documents transactions as signed instructions from accounts, while blocks contain transactions and related execution results. Although transaction scripts, smart-contract calls, or payload fields may specify actions or application data, they do not serve as a universal categorization mechanism for all data in blocks. The accurate conceptual distinction is that transaction formats and block formats structure and record blockchain data according to protocol rules. See the Bitcoin Developer Guide: Transactions and the Ethereum developer documentation on transactions.

QUESTION NO: 10

What does a Merkle tree provide?

A. Efficient transaction verification and tamper-evident data integrity

B. Anonymity and transparency

C. A Turing complete distributed network

D. A way to deploy smart contracts onto the Blockchain

E. All of the above

ANSWER: A

Explanation:

Efficient transaction verification and tamper-evident data integrity is correct because a Merkle tree organizes transaction hashes into a hierarchy ending in one Merkle root. A block header commits to this root, so the root acts as a compact cryptographic summary of all transactions in that block. To verify that a particular transaction belongs in the block, a node does not need to download or process every transaction. Instead, it can receive the transaction and a short Merkle proof consisting of the sibling hashes along the path to the root. The verifier repeatedly hashes these values and checks whether the resulting root matches the root committed in the block header.

This structure makes inclusion checks efficient, requiring a number of hashes that grows logarithmically with the number of transactions. It is also tamper-evident: changing a transaction changes its leaf hash and consequently changes the Merkle root, exposing that the committed data no longer matches. Bitcoin uses a Merkle root in every block header, enabling simplified payment verification clients to validate transaction inclusion using Merkle branches. See the Bitcoin Developer Guide's block-chain reference and Bitcoin's white paper.