基础知识类6-10

面试问答题（中英文）

基础知识类（6-10）

6.What are the main hash functions used in Ether?

以太坊主要使用什么哈希函数?

答：

以太坊主要使用两种哈希函数，分别是 Keccak-256 和 SHA-3。Keccak-256 是以太坊最初采用的哈希函数，它是 SHA-3 标准的一个变种。在以太坊中，Keccak-256 用于计算地址、交易哈希和区块哈希等重要数据的哈希值。SHA-3 是一种新的哈希函数标准，它比 Keccak-256 更加安全和高效。在以太坊中，SHA-3 用于计算合约代码的哈希值。

There are two main hash functions used in Ether, Keccak-256 and SHA-3. Keccak-256 is the original hash function used in Ether, and it is a variant of the SHA-3 standard. In Ether, Keccak-256 is used to calculate hashes of important data such as addresses, transaction hashes, and block hashes.SHA-3 is a new hash function standard that is more secure and efficient than Keccak-256. In Ether, SHA-3 is used to calculate hashes of contract codes.

7、1 个Ether 相当于 多少个 gwei ?

How many gwei does 1 ether equal?

答：

1 ether = 10^9 Gwei

8、1 个Ether 相当于 多少个 wei ?

How many wei are equivalent to 1 Ether?

答：

1 ether = 10^18 Wei

9、assert 和 require 之间有什么区别?

What is the difference between assert and require?

答:

在 Solidity 中，assert 和 require 都是用于检查条件的函数。当条件不满足时，它们会抛出错误或异常。它们之间的区别在于，require 用于在执行前验证输入和条件，而 assert 用于检查不应该为假的代码，失败的断言可能意味着代码层面存在错误。当我们想在执行之前验证输入和条件时，我们使用 require。当我们想要检查可以而且永远不应该为 false 的代码时，我们使用 assert。如果失败，则意味着存在错误，在这种情况下，我们应该使用 assert。

In Solidity, both assert and require are functions that are used to check conditions. They throw errors or exceptions when conditions are not met. The difference between them is that require is used to validate inputs and conditions before execution, while assert is used to check for code that shouldn't be false, and failed assertions can mean that there is an error at the code level. We use require when we want to validate inputs and conditions before execution, and assert when we want to check for code that can and should never be false, and if it fails, it means there is an error, in which case we should use assert.

10、当在选择 assert 和 require 之间做出决定时，需要考虑的方面:

Aspects to consider when deciding between assert and require:

答:

Gas 优化的效率：如果 assert 返回一个 false 语句，它会编译为 0xfe，这是一个无效的操作码，它会用完所有剩余的 gas，从而完全恢复更改。如果 require 返回错误语句，它会编译为 0xfd，这是 REVERT 的操作码，这意味着它将返回剩余的 gas。

Bytecode 分析：如果你使用 assert，你的代码将会更小，因为它不会返回任何东西。如果你使用 require，你的代码将会更大，因为它需要返回错误信息。

Efficiency of Gas Optimization: If assert returns a false statement, it will compile to 0xfe, which is an invalid opcode, and it will use up all the remaining gas to fully revert the change. If require returns an error statement, it compiles to 0xfd, which is the opcode for REVERT, meaning it will return the remaining gas.

Bytecode analysis: If you use assert, your code will be smaller because it won't return anything. If you use require, your code will be bigger because it needs to return error messages.