The `SimpleStorage` smart contract is a basic example of a Solidity contract that allows users to store and retrieve an unsigned integer value. The contract consists of two main functions: `set` and `get`.

1. `set(uint x)`: This function allows anyone to set the value of `storedData` to the provided unsigned integer `x`. It is a public function, meaning that anyone can call it. When called, it updates the `storedData` variable with the new value, effectively overwriting the previously stored value. This function does not return any value and is purely responsible for changing the state of the contract.

2. `get()`: This function is a view function, which means it can be called by anyone without incurring any gas costs since it only reads data and doesn't modify the contract state. When called, it retrieves and returns the current value stored in `storedData`. It does not require any input parameters and provides a read-only access to the value stored in the contract.

Overall, the `SimpleStorage` contract can be seen as a basic data storage contract, allowing users to set and get an unsigned integer value. It showcases the fundamental concepts of state variables, setter functions (`set`), and getter functions (`get`) in Solidity. Note that this contract is simple and has no access control mechanisms or security features, which would be essential in real-world applications.