# **Smart Contract for Credential Lease Code Explanation**

## Language: Solidity

**Data Structure for Lease**

A white background with black text

Description automatically generated

The Lease struct holds all necessary details about a lease:

|  |  |
| --- | --- |
| Attribute | Purpose |
| leaseID | Unique identifier for the lease |
| ownerDID | Address of the credential owner |
| lesseeDID | Address of the person leasing the credential |
| vcIdentifier | Identifier for the Verifiable Credential |
| expiration | Unix timestamp for when the lease expires |
| revoked | Boolean indicating if the lease has been revoked |

**Function: createLease**

A computer code with black text

Description automatically generated

**Purpose:** This function allows the owner of a credential to create a lease for a lessee, specifying the lease duration.

**Key Steps:**

1. **Input Validation:** Ensures the lease duration is valid.
2. **Calculate Expiration:** Adds the lease duration to the current timestamp to determine when the lease will expire.
3. **Store Lease:** Creates a new Lease struct, populates it with data, and stores it in the leases mapping using a unique leaseID.

**Function: revokeLease**

A computer code with black text

Description automatically generated

A computer screen shot of a computer code

Description automatically generated

**Purpose:** This function allows the owner of a credential to revoke an active lease, making it invalid immediately.

**Key Steps:**

1. **Find Lease:** Fetches the lease from the leases mapping using the provided leaseID.
2. **Ownership Check:** Ensures only the owner of the lease (based on the ownerDID) can revoke it.
3. **Revocation Status Check:** Confirms that the lease hasn’t already been revoked.
4. **Update Status:** Sets the revoked property of the lease to true

**Function: validateLease**

A computer code with text

Description automatically generated

**Purpose:** This function allows anyone (owner, lessee, or verifier) to check if a lease is currently valid.

**Key Steps:**

1. **Fetch Lease:** Retrieves the lease data from the leases mapping using the leaseID.
2. **Revocation Check:** Verifies that the lease hasn’t been revoked.
3. **Expiration Check:** Compares the current time (block.timestamp) to the lease’s expiration time.
4. **Return Status:** Returns true if the lease is valid (not revoked and not expired); otherwise, returns false.