Module sui_system::sui_system_state_inner

A list of system config parameters.

Added min_validator_count.

The top-level object containing all information of the Sui system.

Uses SystemParametersV2 as the parameters.

Event containing system-level epoch information, emitted during the epoch advancement transaction.

Create a new SuiSystemState object and make it shared. This function will be called only once in genesis.

Can be called by anyone who wishes to become a validator candidate and starts accruing delegated stakes in their staking pool. Once they have at least MIN_VALIDATOR_JOINING_STAKE amount of stake they can call request_add_validator to officially become an active validator at the next epoch. Aborts if the caller is already a pending or active validator, or a validator candidate. Note: proof_of_possession MUST be a valid signature using sui_address and protocol_pubkey_bytes. To produce a valid PoP, run [fn test_proof_of_possession].

Called by a validator candidate to remove themselves from the candidacy. After this call their staking pool becomes deactivate.

Called by a validator candidate to add themselves to the active validator set beginning next epoch. Aborts if the validator is a duplicate with one of the pending or active validators, or if the amount of stake the validator has doesn't meet the min threshold, or if the number of new validators for the next epoch has already reached the maximum.

A validator can call this function to request a removal in the next epoch. We use the sender of ctx to look up the validator (i.e. sender must match the sui_address in the validator). At the end of the epoch, the validator object will be returned to the sui_address of the validator.

A validator can call this function to submit a new gas price quote, to be used for the reference gas price calculation at the end of the epoch.

This function is used to set new gas price for candidate validators

A validator can call this function to set a new commission rate, updated at the end of the epoch.

This function is used to set new commission rate for candidate validators

Add stake to a validator's staking pool.

Add stake to a validator's staking pool using multiple coins.

Withdraw some portion of a stake from a validator's staking pool.

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

Undo a report_validator action. Aborts if

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

Update a validator's name.

Update a validator's description

Update a validator's image url

Update a validator's project url

Update a validator's network address. The change will only take effects starting from the next epoch.

Update candidate validator's network address.

Update a validator's p2p address. The change will only take effects starting from the next epoch.

Update candidate validator's p2p address.

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

Update candidate validator's narwhal primary address.

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

Update candidate validator's narwhal worker address.

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

Update candidate validator's public key of protocol key and proof of possession.

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

Update candidate validator's public key of worker key.

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

Update candidate validator's public key of network key.

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

Returns unix timestamp of the start of current epoch

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

Returns reference to the staking pool mappings that map pool ids to active validator addresses

Returns all the validators who are currently reporting addr

Extract required Balance from vector of Coin , transfer the remainder back to sender.

Struct

A list of system config parameters.

```bash
```

Added min_validator_count.

```bash
```

The top-level object containing all information of the Sui system.

```bash
```

Uses SystemParametersV2 as the parameters.

```bash
```

Event containing system-level epoch information, emitted during the epoch advancement transaction.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new SuiSystemState object and make it shared. This function will be called only once in genesis.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Can be called by anyone who wishes to become a validator candidate and starts accruing delegated stakes in their staking pool. Once they have at least MIN_VALIDATOR_JOINING_STAKE amount of stake they can call request_add_validator to officially become an active validator at the next epoch. Aborts if the caller is already a pending or active validator, or a validator candidate. Note: proof_of_possession MUST be a valid signature using sui_address and protocol_pubkey_bytes. To produce a valid PoP, run [fn test_proof_of_possession].

```bash
bash (package)
```

```bash
bash (package)
```

Called by a validator candidate to remove themselves from the candidacy. After this call their staking pool becomes deactivate.

```bash
bash (package)
```

```bash
bash (package)
```

Called by a validator candidate to add themselves to the active validator set beginning next epoch. Aborts if the validator is a duplicate with one of the pending or active validators, or if the amount of stake the validator has doesn't meet the min threshold, or if the number of new validators for the next epoch has already reached the maximum.

```bash
bash (package)
```

```bash
bash (package)
```

A validator can call this function to request a removal in the next epoch. We use the sender of ctx to look up the validator (i.e. sender must match the sui_address in the validator). At the end of the epoch, the validator object will be returned to the sui_address of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

A validator can call this function to submit a new gas price quote, to be used for the reference gas price calculation at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

This function is used to set new gas price for candidate validators

```bash
bash (package)
```

```bash
bash (package)
```

A validator can call this function to set a new commission rate, updated at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

This function is used to set new commission rate for candidate validators

```bash
bash (package)
```

```bash
bash (package)
```

Add stake to a validator's staking pool.

```bash
bash (package)
```

```bash
bash (package)
```

Add stake to a validator's staking pool using multiple coins.

```bash
bash (package)
```

```bash
bash (package)
```

Withdraw some portion of a stake from a validator's staking pool.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash
bash (package)
```

```bash
bash (package)
```

Undo a report_validator action. Aborts if

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's name.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's description

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's image url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's project url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's network address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's p2p address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal primary address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal worker address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of worker key.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of network key.

```bash
bash (package)
```

```bash
bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
bash (package)
```

```bash
bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
bash (package)
```

```bash
bash (package)
```

Returns unix timestamp of the start of current epoch

```bash
bash (package)
```

```bash
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
(package)
```

```bash
(package)
```

Returns all the validators who are currently reporting addr

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

```bash
```

```bash
```

Struct

Added min_validator_count.

```bash
```

The top-level object containing all information of the Sui system.

```bash
```

Uses SystemParametersV2 as the parameters.

```bash
```

Event containing system-level epoch information, emitted during the epoch advancement transaction.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new SuiSystemState object and make it shared. This function will be called only once in genesis.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Can be called by anyone who wishes to become a validator candidate and starts accruing delegated stakes in their staking pool. Once they have at least MIN_VALIDATOR_JOINING_STAKE amount of stake they can call request_add_validator to officially become an active validator at the next epoch. Aborts if the caller is already a pending or active validator, or a validator candidate. Note: proof_of_possession MUST be a valid signature using sui_address and protocol_pubkey_bytes. To produce a valid PoP, run [fn test_proof_of_possession].

```bash
bash (package)
```

```bash
bash (package)
```

Called by a validator candidate to remove themselves from the candidacy. After this call their staking pool becomes deactivate.

```bash
bash (package)
```

```bash
bash (package)
```

Called by a validator candidate to add themselves to the active validator set beginning next epoch. Aborts if the validator is a duplicate with one of the pending or active validators, or if the amount of stake the validator has doesn't meet the min threshold, or if the number of new validators for the next epoch has already reached the maximum.

```bash
bash (package)
```

```bash
bash (package)
```

A validator can call this function to request a removal in the next epoch. We use the sender of ctx to look up the validator (i.e. sender must match the sui_address in the validator). At the end of the epoch, the validator object will be returned to the sui_address of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

A validator can call this function to submit a new gas price quote, to be used for the reference gas price calculation at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

This function is used to set new gas price for candidate validators

```bash
bash (package)
```

```bash
bash (package)
```

A validator can call this function to set a new commission rate, updated at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

This function is used to set new commission rate for candidate validators
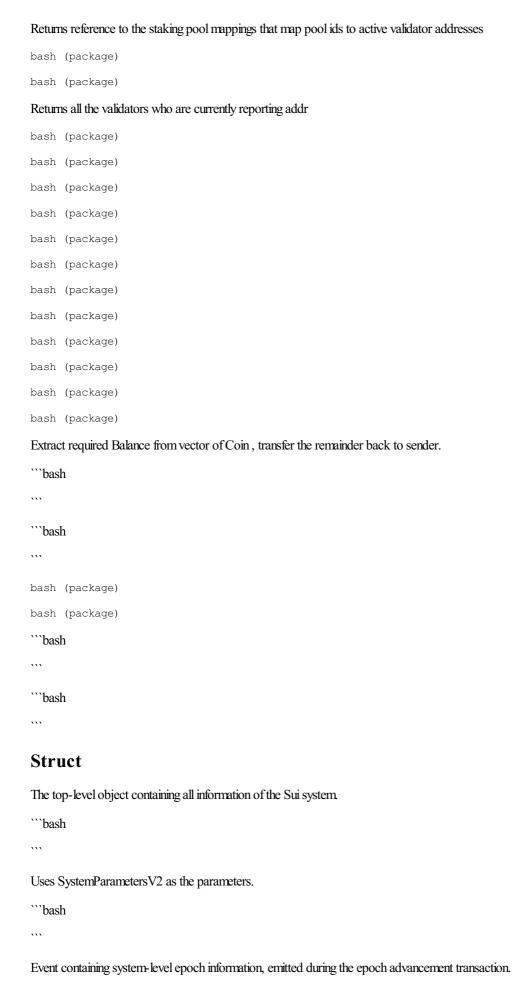
```bash
bash (package)
```

```bash
bash (package)
```

Add stake to a validator's staking pool.

```bash
bash (package)
```

```bash
bash (package)
```

Add stake to a validator's staking pool using multiple coins.

```bash
bash (package)
```

```bash
bash (package)
```

Withdraw some portion of a stake from a validator's staking pool.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash
bash (package)
```

```bash
bash (package)
```

Undo a report_validator action. Aborts if

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's name.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's description

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's image url

```
bash (package)
```

```
bash (package)
```

Update a validator's project url

```
bash (package)
```

```
bash (package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's network address.

```
bash (package)
```

```
bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's p2p address.

```
bash (package)
```

```
bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's narwhal primary address.

```
bash (package)
```

```
bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's narwhal worker address.

```
bash (package)
```

```
bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```
bash (package)
```

```
bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of worker key.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of network key.

```bash
bash (package)
```

```bash
bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
bash (package)
```

```bash
bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
bash (package)
```

```bash
bash (package)
```

Returns unix timestamp of the start of current epoch

```bash
bash (package)
```

```bash
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
bash (package)
```

```bash
bash (package)
```

Returns all the validators who are currently reporting addr

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

Struct

The top-level object containing all information of the Sui system.

```bash
```

Uses SystemParametersV2 as the parameters.

```bash
```

Event containing system-level epoch information, emitted during the epoch advancement transaction.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new SuiSystemState object and make it shared. This function will be called only once in genesis.

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```bash (package)```

```bash (package)```

```bash (package)```

Can be called by anyone who wishes to become a validator candidate and starts accruing delegated stakes in their staking pool. Once they have at least MIN_VALIDATOR_JOINING_STAKE amount of stake they can call request_add_validator to officially become an active validator at the next epoch. Aborts if the caller is already a pending or active validator, or a validator candidate. Note: proof_of_possession MUST be a valid signature using sui_address and protocol_pubkey_bytes. To produce a valid PoP, run [fn test_proof_of_possession].

```bash (package)```

```bash (package)```

Called by a validator candidate to remove themselves from the candidacy. After this call their staking pool becomes deactivate.

```bash (package)```

```bash (package)```

Called by a validator candidate to add themselves to the active validator set beginning next epoch. Aborts if the validator is a duplicate with one of the pending or active validators, or if the amount of stake the validator has doesn't meet the min threshold, or if the number of new validators for the next epoch has already reached the maximum.

```bash (package)```

```bash (package)```

A validator can call this function to request a removal in the next epoch. We use the sender of ctx to look up the validator (i.e. sender must match the sui_address in the validator). At the end of the epoch, the validator object will be returned to the sui_address of the validator.

```bash (package)```

```bash (package)```

A validator can call this function to submit a new gas price quote, to be used for the reference gas price calculation at the end of the epoch.

```bash (package)```

```bash (package)```

This function is used to set new gas price for candidate validators

```bash (package)```

```bash (package)```

A validator can call this function to set a new commission rate, updated at the end of the epoch.

```bash (package)```

```bash (package)```

This function is used to set new commission rate for candidate validators

```bash (package)```

```bash (package)```

Add stake to a validator's staking pool.

```bash (package)```

```bash (package)```

Add stake to a validator's staking pool using multiple coins.

```bash
bash (package)
```

```bash
bash (package)
```

Withdraw some portion of a stake from a validator's staking pool.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash
bash (package)
```

```bash
bash (package)
```

Undo a report_validator action. Aborts if

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's name.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's description

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's image url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's project url

```
bash (package)
```

```
bash (package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's network address.

```
bash (package)
```

```
bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's p2p address.

```
bash (package)
```

```
bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's narwhal primary address.

```
bash (package)
```

```
bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's narwhal worker address.

```
bash (package)
```

```
bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```
bash (package)
```

```
bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's public key of worker key.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of network key.

```bash
bash (package)
```

```bash
bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
bash (package)
```

```bash
bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
bash (package)
```

```bash
bash (package)
```

Returns unix timestamp of the start of current epoch

```bash
bash (package)
```

```bash
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
bash (package)
```

```bash
bash (package)
```

Returns all the validators who are currently reporting addr

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

```bash
```

```bash
```

Struct

Uses SystemParametersV2 as the parameters.

```bash
```

Event containing system-level epoch information, emitted during the epoch advancement transaction.

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new SuiSystemState object and make it shared. This function will be called only once in genesis.

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Can be called by anyone who wishes to become a validator candidate and starts accruing delegated stakes in their staking pool. Once they have at least MIN_VALIDATOR_JOINING_STAKE amount of stake they can call request_add_validator to officially become an active validator at the next epoch. Aborts if the caller is already a pending or active validator, or a validator candidate. Note: proof_of_possession MUST be a valid signature using sui_address and protocol_pubkey_bytes. To produce a valid PoP, run [fn test_proof_of_possession].

```bash
(package)
```

```bash
(package)
```

Called by a validator candidate to remove themselves from the candidacy. After this call their staking pool becomes deactivate.

```bash
(package)
```

```bash
(package)
```

Called by a validator candidate to add themselves to the active validator set beginning next epoch. Aborts if the validator is a duplicate with one of the pending or active validators, or if the amount of stake the validator has doesn't meet the min threshold, or if the number of new validators for the next epoch has already reached the maximum.

```bash
(package)
```

```bash
(package)
```

A validator can call this function to request a removal in the next epoch. We use the sender of ctx to look up the validator (i.e. sender must match the sui_address in the validator). At the end of the epoch, the validator object will be returned to the sui_address of the validator.

```bash
(package)
```

```bash
(package)
```

A validator can call this function to submit a new gas price quote, to be used for the reference gas price calculation at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

This function is used to set new gas price for candidate validators

```bash
(package)
```

```bash
(package)
```

A validator can call this function to set a new commission rate, updated at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

This function is used to set new commission rate for candidate validators

```bash
(package)
```

```bash
(package)
```

Add stake to a validator's staking pool.

```bash
(package)
```

```bash
(package)
```

Add stake to a validator's staking pool using multiple coins.

```bash
(package)
```

```bash
(package)
```

Withdraw some portion of a stake from a validator's staking pool.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash
(package)
```

```bash
(package)
```

Undo a report_validator action. Aborts if

```bash
(package)
```

```bash
(package)
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```bash
(package)
```

```bash
(package)
```

Update a validator's name.

```bash
(package)
```

```bash
(package)
```

Update a validator's description

```bash
(package)
```

```bash
(package)
```

Update a validator's image url

```bash
(package)
```

```bash
(package)
```

Update a validator's project url

```bash
(package)
```

```bash
(package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's network address.

```bash
(package)
```

```bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's p2p address.

```bash (package)
```

```bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's narwhal primary address.

```bash (package)
```

```bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's narwhal worker address.

```bash (package)
```

```bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash (package)
```

```bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of worker key.

```bash (package)
```

```bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of network key.

```bash (package)
```

```bash
bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
bash (package)
```

```bash
bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
bash (package)
```

```bash
bash (package)
```

Returns unix timestamp of the start of current epoch

```bash
bash (package)
```

```bash
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
bash (package)
```

```bash
bash (package)
```

Returns all the validators who are currently reporting addr

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash

```

```bash

```

```bash
bash (package)
```

```bash
bash (package)
```

```bash

```

```bash

```

Struct

Event containing system-level epoch information, emitted during the epoch advancement transaction.

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```bash

```

```
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new SuiSystemState object and make it shared. This function will be called only once in genesis.

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Can be called by anyone who wishes to become a validator candidate and starts accruing delegated stakes in their staking pool. Once they have at least MIN_VALIDATOR_JOINING_STAKE amount of stake they can call request_add_validator to officially become an active validator at the next epoch. Aborts if the caller is already a pending or active validator, or a validator candidate. Note: proof_of_possession MUST be a valid signature using sui_address and protocol_pubkey_bytes. To produce a valid PoP, run [fn test_proof_of_possession].

```
bash (package)
```

```
bash (package)
```

Called by a validator candidate to remove themselves from the candidacy. After this call their staking pool becomes deactivate.

```
bash (package)
```

```
bash (package)
```

Called by a validator candidate to add themselves to the active validator set beginning next epoch. Aborts if the validator is a duplicate with one of the pending or active validators, or if the amount of stake the validator has doesn't meet the min threshold, or if the number of new validators for the next epoch has already reached the maximum.

```
bash (package)
```

```
bash (package)
```

A validator can call this function to request a removal in the next epoch. We use the sender of ctx to look up the validator (i.e.

sender must match the sui_address in the validator). At the end of the epoch, the [validator](#) object will be returned to the sui_address of the validator.

```bash
(package)
```

```bash
(package)
```

A validator can call this function to submit a new gas price quote, to be used for the reference gas price calculation at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

This function is used to set new gas price for candidate validators

```bash
(package)
```

```bash
(package)
```

A validator can call this function to set a new commission rate, updated at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

This function is used to set new commission rate for candidate validators

```bash
(package)
```

```bash
(package)
```

Add stake to a validator's staking pool.

```bash
(package)
```

```bash
(package)
```

Add stake to a validator's staking pool using multiple coins.

```bash
(package)
```

```bash
(package)
```

Withdraw some portion of a stake from a validator's staking pool.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash
(package)
```

```bash
(package)
```

Undo a [report_validator](#) action. Aborts if

```bash
(package)
```

```bash
(package)
```

```bash

```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

`bash (package)`

`bash (package)`

Update a validator's name.

`bash (package)`

`bash (package)`

Update a validator's description

`bash (package)`

`bash (package)`

Update a validator's image url

`bash (package)`

`bash (package)`

Update a validator's project url

`bash (package)`

`bash (package)`

Update a validator's network address. The change will only take effects starting from the next epoch.

`bash (package)`

`bash (package)`

Update candidate validator's network address.

`bash (package)`

`bash (package)`

Update a validator's p2p address. The change will only take effects starting from the next epoch.

`bash (package)`

`bash (package)`

Update candidate validator's p2p address.

`bash (package)`

`bash (package)`

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

`bash (package)`

`bash (package)`

Update candidate validator's narwhal primary address.

```bash
(package)
```

```bash
(package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's narwhal worker address.

```bash
(package)
```

```bash
(package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash
(package)
```

```bash
(package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of worker key.

```bash
(package)
```

```bash
(package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of network key.

```bash
(package)
```

```bash
(package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
(package)
```

```bash
(package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
(package)
```

```bash
(package)
```

Returns unix timestamp of the start of current epoch

```bash
(package)
```

```bash
(package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
(package)
```

```bash
(package)
```

Returns all the validators who are currently reporting addr

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```
bash (package)
bash (package)
```

```bash
```

```bash
```

Constants

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash
```

```bash

```

Create a new SuiSystemState object and make it shared. This function will be called only once in genesis.

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Can be called by anyone who wishes to become a validator candidate and starts accruing delegated stakes in their staking pool. Once they have at least MIN_VALIDATOR_JOINING_STAKE amount of stake they can call request_add_validator to officially become an active validator at the next epoch. Aborts if the caller is already a pending or active validator, or a validator candidate. Note: proof_of_possession MUST be a valid signature using sui_address and protocol_pubkey_bytes. To produce a valid PoP, run [fn test_proof_of_possession].

```
bash (package)
```

```
bash (package)
```

Called by a validator candidate to remove themselves from the candidacy. After this call their staking pool becomes deactivate.

```
bash (package)
```

```
bash (package)
```

Called by a validator candidate to add themselves to the active validator set beginning next epoch. Aborts if the validator is a duplicate with one of the pending or active validators, or if the amount of stake the validator has doesn't meet the min threshold, or if the number of new validators for the next epoch has already reached the maximum.

```
bash (package)
```

```
bash (package)
```

A validator can call this function to request a removal in the next epoch. We use the sender of ctx to look up the validator (i.e. sender must match the sui_address in the validator). At the end of the epoch, the validator object will be returned to the sui_address of the validator.

```
bash (package)
```

```
bash (package)
```

A validator can call this function to submit a new gas price quote, to be used for the reference gas price calculation at the end of the epoch.

```
bash (package)
```

```
bash (package)
```

This function is used to set new gas price for candidate validators

```
bash (package)
```

```
bash (package)
```

A validator can call this function to set a new commission rate, updated at the end of the epoch.

```
bash (package)
```

```
bash (package)
```

This function is used to set new commission rate for candidate validators

```bash
(package)
```

```bash
(package)
```

Add stake to a validator's staking pool.

```bash
(package)
```

```bash
(package)
```

Add stake to a validator's staking pool using multiple coins.

```bash
(package)
```

```bash
(package)
```

Withdraw some portion of a stake from a validator's staking pool.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash
(package)
```

```bash
(package)
```

Undo a report_validator action. Aborts if

```bash
(package)
```

```bash
(package)
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```bash
(package)
```

```bash
(package)
```

Update a validator's name.

```bash
(package)
```

```bash
(package)
```

Update a validator's description

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's image url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's project url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's network address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's p2p address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal primary address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal worker address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of worker key.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of network key.

```bash
bash (package)
```

```bash
bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
bash (package)
```

```bash
bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
bash (package)
```

```bash
bash (package)
```

Returns unix timestamp of the start of current epoch

```bash
bash (package)
```

```bash
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
(package)
```

```bash
(package)
```

Returns all the validators who are currently reporting addr

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Extract required Balance from vector of Coin, transfer the remainder back to sender.

```bash

```

```bash

```

```bash
(package)
```

```bash
(package)
```

```bash

```

```bash

```

Function

Create a new SuiSystemState object and make it shared. This function will be called only once in genesis.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Can be called by anyone who wishes to become a validator candidate and starts accruing delegated stakes in their staking pool. Once they have at least MIN_VALIDATOR_JOINING_STAKE amount of stake they can call request_add_validator to officially become an active validator at the next epoch. Aborts if the caller is already a pending or active validator, or a validator candidate. Note: proof_of_possession MUST be a valid signature using sui_address and protocol_pubkey_bytes. To produce a valid PoP, run [fn test_proof_of_possession].

```bash
bash (package)
```

```bash
bash (package)
```

Called by a validator candidate to remove themselves from the candidacy. After this call their staking pool becomes deactivate.

```bash
bash (package)
```

```bash
bash (package)
```

Called by a validator candidate to add themselves to the active validator set beginning next epoch. Aborts if the validator is a duplicate with one of the pending or active validators, or if the amount of stake the validator has doesn't meet the min threshold, or if the number of new validators for the next epoch has already reached the maximum.

```bash
bash (package)
```

```bash
bash (package)
```

A validator can call this function to request a removal in the next epoch. We use the sender of ctx to look up the validator (i.e. sender must match the sui_address in the validator). At the end of the epoch, the validator object will be returned to the sui_address of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

A validator can call this function to submit a new gas price quote, to be used for the reference gas price calculation at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

This function is used to set new gas price for candidate validators

```bash
bash (package)
```

```bash
bash (package)
```

A validator can call this function to set a new commission rate, updated at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

This function is used to set new commission rate for candidate validators

```bash
bash (package)
```

```bash
bash (package)
```

Add stake to a validator's staking pool.

```bash
bash (package)
```

```bash
bash (package)
```

Add stake to a validator's staking pool using multiple coins.

```bash
bash (package)
```

```bash
(package)
```

Withdraw some portion of a stake from a validator's staking pool.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash
(package)
```

```bash
(package)
```

Undo a report_validator action. Aborts if

```bash
(package)
```

```bash
(package)
```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```bash
(package)
```

```bash
(package)
```

Update a validator's name.

```bash
(package)
```

```bash
(package)
```

Update a validator's description

```bash
(package)
```

```bash
(package)
```

Update a validator's image url

```bash
(package)
```

```bash
(package)
```

Update a validator's project url

```bash
(package)
```

```bash
bash (package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's network address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's p2p address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal primary address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal worker address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of worker key.

```bash
bash (package)
```

```bash
(package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of network key.

```bash
(package)
```

```bash
(package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
(package)
```

```bash
(package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
(package)
```

```bash
(package)
```

Returns unix timestamp of the start of current epoch

```bash
(package)
```

```bash
(package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
(package)
```

```bash
(package)
```

Returns all the validators who are currently reporting addr

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

```bash
```

```bash
```

Function

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Can be called by anyone who wishes to become a validator candidate and starts accruing delegated stakes in their staking pool. Once they have at least MIN_VALIDATOR_JOINING_STAKE amount of stake they can call request_add_validator to officially become an active validator at the next epoch. Aborts if the caller is already a pending or active validator, or a validator candidate. Note: proof_of_possession MUST be a valid signature using sui_address and protocol_pubkey_bytes. To produce a valid PoP, run [fn test_proof_of_possession].

```bash
(package)
```

```bash
(package)
```

Called by a validator candidate to remove themselves from the candidacy. After this call their staking pool becomes deactivate.

```bash
(package)
```

```bash
(package)
```

Called by a validator candidate to add themselves to the active validator set beginning next epoch. Aborts if the validator is a duplicate with one of the pending or active validators, or if the amount of stake the validator has doesn't meet the min threshold, or if the number of new validators for the next epoch has already reached the maximum.

```bash
bash (package)
```

```bash
bash (package)
```

A validator can call this function to request a removal in the next epoch. We use the sender of ctx to look up the validator (i.e. sender must match the sui_address in the validator). At the end of the epoch, the validator object will be returned to the sui_address of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

A validator can call this function to submit a new gas price quote, to be used for the reference gas price calculation at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

This function is used to set new gas price for candidate validators

```bash
bash (package)
```

```bash
bash (package)
```

A validator can call this function to set a new commission rate, updated at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

This function is used to set new commission rate for candidate validators

```bash
bash (package)
```

```bash
bash (package)
```

Add stake to a validator's staking pool.

```bash
bash (package)
```

```bash
bash (package)
```

Add stake to a validator's staking pool using multiple coins.

```bash
bash (package)
```

```bash
bash (package)
```

Withdraw some portion of a stake from a validator's staking pool.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash
bash (package)
```

```bash
bash (package)
```

Undo a report_validator action. Aborts if

```bash
(package)
```

```bash
(package)
```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```bash
(package)
```

```bash
(package)
```

Update a validator's name.

```bash
(package)
```

```bash
(package)
```

Update a validator's description

```bash
(package)
```

```bash
(package)
```

Update a validator's image url

```bash
(package)
```

```bash
(package)
```

Update a validator's project url

```bash
(package)
```

```bash
(package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's network address.

```bash
(package)
```

```bash
(package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's p2p address.

```bash
(package)
```

```bash
(package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's narwhal primary address.

```bash
(package)
```

```bash
(package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's narwhal worker address.

```bash
(package)
```

```bash
(package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash
(package)
```

```bash
(package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of worker key.

```bash
(package)
```

```bash
(package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of network key.

```bash
(package)
```

```bash
(package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
(package)
```

```bash
(package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
bash (package)
```

```bash
bash (package)
```

Returns unix timestamp of the start of current epoch

```bash
bash (package)
```

```bash
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
bash (package)
```

```bash
bash (package)
```

Returns all the validators who are currently reporting addr

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash

```

```bash

```

```bash
bash (package)
```

```bash
bash (package)
```

```bash

```

```bash

```

Function

```bash
bash (package)
```

```bash
bash (package)
```

Can be called by anyone who wishes to become a validator candidate and starts accruing delegated stakes in their staking pool. Once they have at least MIN_VALIDATOR_JOINING_STAKE amount of stake they can call request_add_validator to officially become an active validator at the next epoch. Aborts if the caller is already a pending or active validator, or a validator candidate. Note: proof_of_possession MUST be a valid signature using sui_address and protocol_pubkey_bytes. To produce a valid PoP, run [fn test_proof_of_possession].

```bash
bash (package)
```

```bash
bash (package)
```

Called by a validator candidate to remove themselves from the candidacy. After this call their staking pool becomes deactivate.

```bash
bash (package)
```

```bash
bash (package)
```

Called by a validator candidate to add themselves to the active validator set beginning next epoch. Aborts if the validator is a duplicate with one of the pending or active validators, or if the amount of stake the validator has doesn't meet the min threshold, or if the number of new validators for the next epoch has already reached the maximum.

```bash
bash (package)
```

```bash
bash (package)
```

A validator can call this function to request a removal in the next epoch. We use the sender of ctx to look up the validator (i.e. sender must match the sui_address in the validator). At the end of the epoch, the validator object will be returned to the sui_address of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

A validator can call this function to submit a new gas price quote, to be used for the reference gas price calculation at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

This function is used to set new gas price for candidate validators

```bash
(package)
```

```bash
(package)
```

A validator can call this function to set a new commission rate, updated at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

This function is used to set new commission rate for candidate validators

```bash
(package)
```

```bash
(package)
```

Add stake to a validator's staking pool.

```bash
(package)
```

```bash
(package)
```

Add stake to a validator's staking pool using multiple coins.

```bash
(package)
```

```bash
(package)
```

Withdraw some portion of a stake from a validator's staking pool.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash
(package)
```

```bash
(package)
```

Undo a report_validator action. Aborts if

```bash
(package)
```

```bash
(package)
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```bash (package)
```

```bash (package)
```

Update a validator's name.

```bash (package)
```

```bash (package)
```

Update a validator's description

```bash (package)
```

```bash (package)
```

Update a validator's image url

```bash (package)
```

```bash (package)
```

Update a validator's project url

```bash (package)
```

```bash (package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's network address.

```bash (package)
```

```bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's p2p address.

```bash (package)
```

```bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's narwhal primary address.

```bash (package)
```

```bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's narwhal worker address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of worker key.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of network key.

```bash
bash (package)
```

```bash
bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
bash (package)
```

```bash
bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
bash (package)
```

```bash
bash (package)
```

Returns unix timestamp of the start of current epoch

```bash
bash (package)
```

```bash
(package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
(package)
```

```bash
(package)
```

Returns all the validators who are currently reporting addr

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash

```

```bash

```

```bash
(package)
```

```bash
(package)
```

```bash

```

```bash

```

Function

Can be called by anyone who wishes to become a validator candidate and starts accruing delegated stakes in their staking pool. Once they have at least MIN_VALIDATOR_JOINING_STAKE amount of stake they can call request_add_validator to officially become an active validator at the next epoch. Aborts if the caller is already a pending or active validator, or a validator candidate. Note: proof_of_possession MUST be a valid signature using sui_address and protocol_pubkey_bytes. To produce a valid PoP, run [fn test_proof_of_possession].

```bash (package)
```

```bash (package)
```

Called by a validator candidate to remove themselves from the candidacy. After this call their staking pool becomes deactivate.

```bash (package)
```

```bash (package)
```

Called by a validator candidate to add themselves to the active validator set beginning next epoch. Aborts if the validator is a duplicate with one of the pending or active validators, or if the amount of stake the validator has doesn't meet the min threshold, or if the number of new validators for the next epoch has already reached the maximum.

```bash (package)
```

```bash (package)
```

A validator can call this function to request a removal in the next epoch. We use the sender of ctx to look up the validator (i.e. sender must match the sui_address in the validator). At the end of the epoch, the validator object will be returned to the sui_address of the validator.

```bash (package)
```

```bash (package)
```

A validator can call this function to submit a new gas price quote, to be used for the reference gas price calculation at the end of the epoch.

```bash (package)
```

```bash (package)
```

This function is used to set new gas price for candidate validators

```bash (package)
```

```bash (package)
```

A validator can call this function to set a new commission rate, updated at the end of the epoch.

```bash (package)
```

```bash (package)
```

This function is used to set new commission rate for candidate validators

```bash (package)
```

```bash (package)
```

Add stake to a validator's staking pool.

```bash (package)
```

```bash (package)
```

Add stake to a validator's staking pool using multiple coins.

```bash (package)
```

```bash
(package)
```

Withdraw some portion of a stake from a validator's staking pool.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash
(package)
```

```bash
(package)
```

Undo a report_validator action. Aborts if

```bash
(package)
```

```bash
(package)
```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```bash
(package)
```

```bash
(package)
```

Update a validator's name.

```bash
(package)
```

```bash
(package)
```

Update a validator's description

```bash
(package)
```

```bash
(package)
```

Update a validator's image url

```bash
(package)
```

```bash
(package)
```

Update a validator's project url

```bash
(package)
```

```bash
bash (package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's network address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's p2p address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal primary address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal worker address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of worker key.

```bash
bash (package)
```

```bash
(package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of network key.

```bash
(package)
```

```bash
(package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
(package)
```

```bash
(package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
(package)
```

```bash
(package)
```

Returns unix timestamp of the start of current epoch

```bash
(package)
```

```bash
(package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
(package)
```

```bash
(package)
```

Returns all the validators who are currently reporting addr

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash

```

```bash

```

```
bash (package)
```

```
bash (package)
```

```bash

```

```bash

```

Function

Called by a validator candidate to remove themselves from the candidacy. After this call their staking pool becomes deactivate.

```
bash (package)
```

```
bash (package)
```

Called by a validator candidate to add themselves to the active validator set beginning next epoch. Aborts if the validator is a duplicate with one of the pending or active validators, or if the amount of stake the validator has doesn't meet the min threshold, or if the number of new validators for the next epoch has already reached the maximum.

```
bash (package)
```

```
bash (package)
```

A validator can call this function to request a removal in the next epoch. We use the sender of ctx to look up the validator (i.e. sender must match the sui_address in the validator). At the end of the epoch, the validator object will be returned to the sui_address of the validator.

```
bash (package)
```

```
bash (package)
```

A validator can call this function to submit a new gas price quote, to be used for the reference gas price calculation at the end of the

epoch.

```bash
(package)
```

```bash
(package)
```

This function is used to set new gas price for candidate validators

```bash
(package)
```

```bash
(package)
```

A validator can call this function to set a new commission rate, updated at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

This function is used to set new commission rate for candidate validators

```bash
(package)
```

```bash
(package)
```

Add stake to a validator's staking pool.

```bash
(package)
```

```bash
(package)
```

Add stake to a validator's staking pool using multiple coins.

```bash
(package)
```

```bash
(package)
```

Withdraw some portion of a stake from a validator's staking pool.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash
(package)
```

```bash
(package)
```

Undo a report_validator action. Aborts if

```bash
(package)
```

```bash
(package)
```

```bash
```

```bash
```

```bash
```

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```
bash (package)
```

```
bash (package)
```

Update a validator's name.

```
bash (package)
```

```
bash (package)
```

Update a validator's description

```
bash (package)
```

```
bash (package)
```

Update a validator's image url

```
bash (package)
```

```
bash (package)
```

Update a validator's project url

```
bash (package)
```

```
bash (package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's network address.

```
bash (package)
```

```
bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's p2p address.

```
bash (package)
```

```
bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's narwhal primary address.

```
bash (package)
```

```
bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's narwhal worker address.

```bash (package)
```

```bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash (package)
```

```bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of worker key.

```bash (package)
```

```bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of network key.

```bash (package)
```

```bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash (package)
```

```bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
(package)
```

```bash
(package)
```

Returns unix timestamp of the start of current epoch

```bash
(package)
```

```bash
(package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
(package)
```

```bash
(package)
```

Returns all the validators who are currently reporting addr

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Extract required Balance from vector of Coin, transfer the remainder back to sender.

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

```bash

```

```bash

```

Function

Called by a validator candidate to add themselves to the active validator set beginning next epoch. Aborts if the validator is a duplicate with one of the pending or active validators, or if the amount of stake the validator has doesn't meet the min threshold, or if the number of new validators for the next epoch has already reached the maximum.

```bash
 (package)
```

```bash
 (package)
```

A validator can call this function to request a removal in the next epoch. We use the sender of ctx to look up the validator (i.e. sender must match the sui_address in the validator). At the end of the epoch, the [validator](#) object will be returned to the sui_address of the validator.

```bash
 (package)
```

```bash
 (package)
```

A validator can call this function to submit a new gas price quote, to be used for the reference gas price calculation at the end of the epoch.

```bash
 (package)
```

```bash
 (package)
```

This function is used to set new gas price for candidate validators

```bash
 (package)
```

```bash
 (package)
```

A validator can call this function to set a new commission rate, updated at the end of the epoch.

```bash
 (package)
```

```bash
 (package)
```

This function is used to set new commission rate for candidate validators

```bash
 (package)
```

```bash
 (package)
```

Add stake to a validator's staking pool.

```bash
 (package)
```

```bash
 (package)
```

Add stake to a validator's staking pool using multiple coins.

```bash
 (package)
```

```bash
 (package)
```

Withdraw some portion of a stake from a validator's staking pool.

```bash
 (package)
```

```bash
 (package)
```

```bash
 (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash
bash (package)
```

```bash
bash (package)
```

Undo a report_validator action. Aborts if

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's name.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's description

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's image url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's project url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's network address.

```
bash (package)
```

```
bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's p2p address.

```
bash (package)
```

```
bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's narwhal primary address.

```
bash (package)
```

```
bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's narwhal worker address.

```
bash (package)
```

```
bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```
bash (package)
```

```
bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's public key of worker key.

```
bash (package)
```

```
bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's public key of network key.

```bash
bash (package)
```

```bash
bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
bash (package)
```

```bash
bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
bash (package)
```

```bash
bash (package)
```

Returns unix timestamp of the start of current epoch

```bash
bash (package)
```

```bash
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
bash (package)
```

```bash
bash (package)
```

Returns all the validators who are currently reporting addr

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

Function

A validator can call this function to request a removal in the next epoch. We use the sender of ctx to look up the validator (i.e. sender must match the sui_address in the validator). At the end of the epoch, the validator object will be returned to the sui_address of the validator.

```bash
bash (package)
```

```bash
bash (package)
```

A validator can call this function to submit a new gas price quote, to be used for the reference gas price calculation at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

This function is used to set new gas price for candidate validators

```bash
bash (package)
```

```bash
bash (package)
```

A validator can call this function to set a new commission rate, updated at the end of the epoch.

```bash
bash (package)
```

```bash
bash (package)
```

This function is used to set new commission rate for candidate validators

```bash
bash (package)
```

```bash
bash (package)
```

Add stake to a validator's staking pool.

```bash
(package)
```

```bash
(package)
```

Add stake to a validator's staking pool using multiple coins.

```bash
(package)
```

```bash
(package)
```

Withdraw some portion of a stake from a validator's staking pool.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash
(package)
```

```bash
(package)
```

Undo a [report_validator](#) action. Aborts if

```bash
(package)
```

```bash
(package)
```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```bash
(package)
```

```bash
(package)
```

Update a validator's name.

```bash
(package)
```

```bash
(package)
```

Update a validator's description

```bash
(package)
```

```bash
(package)
```

Update a validator's image url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's project url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's network address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's p2p address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal primary address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal worker address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash (package)```

```bash (package)```

Update candidate validator's public key of worker key.

```bash (package)```

```bash (package)```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash (package)```

```bash (package)```

Update candidate validator's public key of network key.

```bash (package)```

```bash (package)```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash (package)```

```bash (package)```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash (package)```

```bash (package)```

Returns unix timestamp of the start of current epoch

```bash (package)```

```bash (package)```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)```

```bash (package)```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)```

```bash (package)```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash (package)```

```bash
(package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
(package)
```

```bash
(package)
```

Returns all the validators who are currently reporting addr

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash

```

```bash

```

```bash
(package)
```

```bash
(package)
```

```bash

```

```bash

```

Function

A validator can call this function to submit a new gas price quote, to be used for the reference gas price calculation at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

This function is used to set new gas price for candidate validators

```bash
(package)
```

```bash
(package)
```

A validator can call this function to set a new commission rate, updated at the end of the epoch.

```bash (package)
```

```bash (package)
```

This function is used to set new commission rate for candidate validators

```bash (package)
```

```bash (package)
```

Add stake to a validator's staking pool.

```bash (package)
```

```bash (package)
```

Add stake to a validator's staking pool using multiple coins.

```bash (package)
```

```bash (package)
```

Withdraw some portion of a stake from a validator's staking pool.

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash (package)
```

```bash (package)
```

Undo a report_validator action. Aborts if

```bash (package)
```

```bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```bash (package)
```

```bash (package)
```

Update a validator's name.

```bash
(package)
```

```bash
(package)
```

Update a validator's description

```bash
(package)
```

```bash
(package)
```

Update a validator's image url

```bash
(package)
```

```bash
(package)
```

Update a validator's project url

```bash
(package)
```

```bash
(package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's network address.

```bash
(package)
```

```bash
(package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's p2p address.

```bash
(package)
```

```bash
(package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's narwhal primary address.

```bash
(package)
```

```bash
(package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's narwhal worker address.

```bash
(package)
```

```bash
(package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next

epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash
(package)
```

```bash
(package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of worker key.

```bash
(package)
```

```bash
(package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of network key.

```bash
(package)
```

```bash
(package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
(package)
```

```bash
(package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
(package)
```

```bash
(package)
```

Returns unix timestamp of the start of current epoch

```bash
(package)
```

```bash
(package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
(package)
```

```bash
(package)
```

Returns all the validators who are currently reporting addr

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

```bash
```

```bash
```

Function

This function is used to set new gas price for candidate validators

```bash
(package)
```

```bash
(package)
```

A validator can call this function to set a new commission rate, updated at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

This function is used to set new commission rate for candidate validators

```bash
(package)
```

```bash
(package)
```

Add stake to a validator's staking pool.

```bash
(package)
```

```bash
(package)
```

Add stake to a validator's staking pool using multiple coins.

```bash
(package)
```

```bash
(package)
```

Withdraw some portion of a stake from a validator's staking pool.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash
(package)
```

```bash
(package)
```

Undo a report_validator action. Aborts if

```bash
(package)
```

```bash
(package)
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's name.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's description

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's image url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's project url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's network address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's p2p address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal primary address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal worker address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of worker key.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of network key.

```bash
bash (package)
```

```bash
bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
bash (package)
```

```bash
bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
bash (package)
```

```bash
bash (package)
```

Returns unix timestamp of the start of current epoch

```bash
bash (package)
```

```bash
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)
```

```bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)
```

```bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash (package)
```

```bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash (package)
```

```bash (package)
```

Returns all the validators who are currently reporting addr

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash (package)
```

```bash (package)
```

```bash
```

```bash
```

Function

A validator can call this function to set a new commission rate, updated at the end of the epoch.

```bash
(package)
```

```bash
(package)
```

This function is used to set new commission rate for candidate validators

```bash
(package)
```

```bash
(package)
```

Add stake to a validator's staking pool.

```bash
(package)
```

```bash
(package)
```

Add stake to a validator's staking pool using multiple coins.

```bash
(package)
```

```bash
(package)
```

Withdraw some portion of a stake from a validator's staking pool.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash
(package)
```

```bash
(package)
```

Undo a report_validator action. Aborts if

```bash
(package)
```

```bash
(package)
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```bash
(package)
```

```bash
(package)
```

Update a validator's name.

```bash
(package)
```

```bash
(package)
```

Update a validator's description

```bash
(package)
```

```bash
(package)
```

Update a validator's image url

```bash
(package)
```

```bash
(package)
```

Update a validator's project url

```bash
(package)
```

```bash
(package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's network address.

```bash
(package)
```

```bash
(package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's p2p address.

```bash
(package)
```

```bash
(package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's narwhal primary address.

```bash
(package)
```

```bash
(package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's narwhal worker address.

```bash
(package)
```

```bash
(package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash (package)```

```bash (package)```

Update candidate validator's public key of protocol key and proof of possession.

```bash (package)```

```bash (package)```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash (package)```

```bash (package)```

Update candidate validator's public key of worker key.

```bash (package)```

```bash (package)```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash (package)```

```bash (package)```

Update candidate validator's public key of network key.

```bash (package)```

```bash (package)```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash (package)```

```bash (package)```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash (package)```

```bash (package)```

Returns unix timestamp of the start of current epoch

```bash (package)```

```bash (package)```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
bash (package)
```

```bash
bash (package)
```

Returns all the validators who are currently reporting addr

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

Function

This function is used to set new commission rate for candidate validators

```bash (package)
```

```bash (package)
```

Add stake to a validator's staking pool.

```bash (package)
```

```bash (package)
```

Add stake to a validator's staking pool using multiple coins.

```bash (package)
```

```bash (package)
```

Withdraw some portion of a stake from a validator's staking pool.

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash (package)
```

```bash (package)
```

Undo a report_validator action. Aborts if

```bash (package)
```

```bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```bash (package)
```

```bash (package)
```

Update a validator's name.

```bash (package)
```

```bash (package)
```

Update a validator's description

```bash
(package)
```

```bash
(package)
```

Update a validator's image url

```bash
(package)
```

```bash
(package)
```

Update a validator's project url

```bash
(package)
```

```bash
(package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's network address.

```bash
(package)
```

```bash
(package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's p2p address.

```bash
(package)
```

```bash
(package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's narwhal primary address.

```bash
(package)
```

```bash
(package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's narwhal worker address.

```bash
(package)
```

```bash
(package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash
(package)
```

```bash
(package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of worker key.

```bash
(package)
```

```bash
(package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of network key.

```bash
(package)
```

```bash
(package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
(package)
```

```bash
(package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
(package)
```

```bash
(package)
```

Returns unix timestamp of the start of current epoch

```bash
(package)
```

```bash
(package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
(package)
```

```bash
(package)
```

Returns all the validators who are currently reporting addr

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Extract required Balance from vector of Coin, transfer the remainder back to sender.

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

```bash
```

```bash
```

Function

Add stake to a validator's staking pool.

```bash
(package)
```

```bash
(package)
```

Add stake to a validator's staking pool using multiple coins.

```bash
bash (package)
```

```bash
bash (package)
```

Withdraw some portion of a stake from a validator's staking pool.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash
bash (package)
```

```bash
bash (package)
```

Undo a [report_validator](#) action. Aborts if

```bash
bash (package)
```

```bash
bash (package)
```

```bash

```

```bash

```

```bash

```

```bash

```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's name.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's description

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's image url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's project url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's network address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's p2p address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal primary address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal worker address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of worker key.

```bash (package)
```

```bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of network key.

```bash (package)
```

```bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash (package)
```

```bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash (package)
```

```bash (package)
```

Returns unix timestamp of the start of current epoch

```bash (package)
```

```bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)
```

```bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)
```

```bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash (package)
```

```bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash (package)
```

```bash (package)
```

Returns all the validators who are currently reporting addr

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

Function

Add stake to a validator's staking pool using multiple coins.

```
bash (package)
```

```
bash (package)
```

Withdraw some portion of a stake from a validator's staking pool.

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash
(package)
```

```bash
(package)
```

Undo a [report_validator](#) action. Aborts if

```bash
(package)
```

```bash
(package)
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```bash
(package)
```

```bash
(package)
```

Update a validator's name.

```bash
(package)
```

```bash
(package)
```

Update a validator's description

```bash
(package)
```

```bash
(package)
```

Update a validator's image url

```bash
(package)
```

```bash
(package)
```

Update a validator's project url

```bash
(package)
```

```bash
(package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's network address.

```bash
(package)
```

```bash
(package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
```
(package)

Update candidate validator's p2p address.

```bash
```
(package)

```bash
```
(package)

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash
```
(package)

```bash
```
(package)

Update candidate validator's narwhal primary address.

```bash
```
(package)

```bash
```
(package)

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash
```
(package)

```bash
```
(package)

Update candidate validator's narwhal worker address.

```bash
```
(package)

```bash
```
(package)

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
```
(package)

```bash
```
(package)

Update candidate validator's public key of protocol key and proof of possession.

```bash
```
(package)

```bash
```
(package)

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
```
(package)

```bash
```
(package)

Update candidate validator's public key of worker key.

```bash
```
(package)

```bash
```
(package)

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
```
(package)

```bash
```
(package)

Update candidate validator's public key of network key.

```bash
```
(package)

```bash
```
(package)

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
```
(package)

```bash
bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
bash (package)
```

```bash
bash (package)
```

Returns unix timestamp of the start of current epoch

```bash
bash (package)
```

```bash
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
bash (package)
```

```bash
bash (package)
```

Returns all the validators who are currently reporting addr

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

```bash
```

```bash
```

Function

Withdraw some portion of a stake from a validator's staking pool.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash
(package)
```

```bash
(package)
```

Undo a report_validator action. Aborts if

```bash
(package)
```

```bash
(package)
```

```bash
```

```bash
```

```bash
```

```bash
```

```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```bash (package)```

```bash (package)```

Update a validator's name.

```bash (package)```

```bash (package)```

Update a validator's description

```bash (package)```

```bash (package)```

Update a validator's image url

```bash (package)```

```bash (package)```

Update a validator's project url

```bash (package)```

```bash (package)```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash (package)```

```bash (package)```

Update candidate validator's network address.

```bash (package)```

```bash (package)```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash (package)```

```bash (package)```

Update candidate validator's p2p address.

```bash (package)```

```bash (package)```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash (package)```

```bash (package)```

Update candidate validator's narwhal primary address.

```bash (package)```

```bash (package)```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash (package)```

```bash (package)```

Update candidate validator's narwhal worker address.

```bash (package)
```

```bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash (package)
```

```bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of worker key.

```bash (package)
```

```bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of network key.

```bash (package)
```

```bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash (package)
```

```bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash (package)
```

```bash (package)
```

Returns unix timestamp of the start of current epoch

```bash
bash (package)
```

```bash
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
bash (package)
```

```bash
bash (package)
```

Returns all the validators who are currently reporting addr

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

```
```

Function

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```
bash (package)
```

```
bash (package)
```

Undo a [report_validator](#) action. Aborts if

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```
bash (package)
```

```
bash (package)
```

Update a validator's name.

```
bash (package)
```

```
bash (package)
```

Update a validator's description

```
bash (package)
```

```
bash (package)
```

Update a validator's image url

```
bash (package)
```

```
bash (package)
```

Update a validator's project url

```
bash (package)
```

```
bash (package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's network address.

```bash (package)
```

```bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's p2p address.

```bash (package)
```

```bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's narwhal primary address.

```bash (package)
```

```bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's narwhal worker address.

```bash (package)
```

```bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash (package)
```

```bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of worker key.

```bash (package)
```

```bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of network key.

```bash
bash (package)
```

```bash
bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
bash (package)
```

```bash
bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
bash (package)
```

```bash
bash (package)
```

Returns unix timestamp of the start of current epoch

```bash
bash (package)
```

```bash
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
bash (package)
```

```bash
bash (package)
```

Returns all the validators who are currently reporting addr

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

Function

```bash
bash (package)
```

```bash
bash (package)
```

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash
bash (package)
```

```bash
bash (package)
```

Undo a report_validator action. Aborts if

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```
bash (package)
```

```
bash (package)
```

Update a validator's name.

```
bash (package)
```

```
bash (package)
```

Update a validator's description

```
bash (package)
```

```
bash (package)
```

Update a validator's image url

```
bash (package)
```

```
bash (package)
```

Update a validator's project url

```
bash (package)
```

```
bash (package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's network address.

```
bash (package)
```

```
bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's p2p address.

```
bash (package)
```

```
bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's narwhal primary address.

```
bash (package)
```

```
bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's narwhal worker address.

```bash (package)
```

```bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash (package)
```

```bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of worker key.

```bash (package)
```

```bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of network key.

```bash (package)
```

```bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash (package)
```

```bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash (package)
```

```bash
(package)
```

Returns unix timestamp of the start of current epoch

```bash
(package)
```

```bash
(package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
(package)
```

```bash
(package)
```

Returns all the validators who are currently reporting addr

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

```bash
```

```
```

```bash
```

Function

Report a validator as a bad or non-performant actor in the system. Succeeds if all the following are satisfied:

```bash
(package)
```

```bash
(package)
```

Undo a report_validator action. Aborts if

```bash
(package)
```

```bash
(package)
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```bash
(package)
```

```bash
(package)
```

Update a validator's name.

```bash
(package)
```

```bash
(package)
```

Update a validator's description

```bash
(package)
```

```bash
(package)
```

Update a validator's image url

```bash
(package)
```

```bash
(package)
```

Update a validator's project url

```bash
(package)
```

```bash
(package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
bash (package)
```

Update candidate validator's network address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's p2p address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal primary address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal worker address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of worker key.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
(package)
```

Update candidate validator's public key of network key.

```bash
(package)
```

```bash
(package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
(package)
```

```bash
(package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
(package)
```

```bash
(package)
```

Returns unix timestamp of the start of current epoch

```bash
(package)
```

```bash
(package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
(package)
```

```bash
(package)
```

Returns all the validators who are currently reporting addr

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

```bash
```

```bash
```

Function

Undo a report_validator action. Aborts if

```bash
(package)
```

```bash
(package)
```

```bash
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```bash
(package)
```

```bash
(package)
```

Update a validator's name.

```bash
(package)
```

```bash
(package)
```

Update a validator's description

```bash
(package)
```

```bash
(package)
```

Update a validator's image url

```bash
(package)
```

```bash
(package)
```

Update a validator's project url

```bash
(package)
```

```bash
(package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's network address.

```bash
(package)
```

```bash
(package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's p2p address.

```bash
(package)
```

```bash
(package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's narwhal primary address.

```bash
(package)
```

```bash
(package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's narwhal worker address.

```bash
(package)
```

```bash
(package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next

epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash (package)
```

```bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of worker key.

```bash (package)
```

```bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of network key.

```bash (package)
```

```bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash (package)
```

```bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash (package)
```

```bash (package)
```

Returns unix timestamp of the start of current epoch

```bash (package)
```

```bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)
```

```bash
(package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
(package)
```

```bash
(package)
```

Returns all the validators who are currently reporting addr

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash

```

```bash

```

```bash
(package)
```

```bash
(package)
```

```bash

```

```bash

```

Function

```bash
```

```
```

```bash
```

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

`bash (package)`

`bash (package)`

Update a validator's name.

`bash (package)`

`bash (package)`

Update a validator's description

`bash (package)`

`bash (package)`

Update a validator's image url

`bash (package)`

`bash (package)`

Update a validator's project url

`bash (package)`

`bash (package)`

Update a validator's network address. The change will only take effects starting from the next epoch.

`bash (package)`

`bash (package)`

Update candidate validator's network address.

`bash (package)`

`bash (package)`

Update a validator's p2p address. The change will only take effects starting from the next epoch.

`bash (package)`

`bash (package)`

Update candidate validator's p2p address.

`bash (package)`

`bash (package)`

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

`bash (package)`

```bash
bash (package)
```

Update candidate validator's narwhal primary address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal worker address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of worker key.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of network key.

```bash
bash (package)
```

```bash
bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
bash (package)
```

```bash
bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
bash (package)
```

```bash
bash (package)
```

Returns unix timestamp of the start of current epoch

```bash
bash (package)
```

```bash
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
bash (package)
```

```bash
bash (package)
```

Returns all the validators who are currently reporting addr

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash

```

```bash
```

bash (package)

bash (package)

```bash
```

```bash
```

Function

```bash
```

```bash
```

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

bash (package)

bash (package)

Update a validator's name.

bash (package)

bash (package)

Update a validator's description

bash (package)

bash (package)

Update a validator's image url

bash (package)

bash (package)

Update a validator's project url

bash (package)

bash (package)

Update a validator's network address. The change will only take effects starting from the next epoch.

bash (package)

bash (package)

Update candidate validator's network address.

bash (package)

bash (package)

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's p2p address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal primary address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal worker address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of worker key.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of network key.

```bash
bash (package)
```

```bash
bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash (package)
```

```bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash (package)
```

```bash (package)
```

Returns unix timestamp of the start of current epoch

```bash (package)
```

```bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)
```

```bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)
```

```bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash (package)
```

```bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash (package)
```

```bash (package)
```

Returns all the validators who are currently reporting addr

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

Function

Create a new UnverifiedValidatorOperationCap, transfer it to the validator and registers it. The original object is thus revoked.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's name.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's description

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's image url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's project url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's network address.

```bash
bash (package)
```

```bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's p2p address.

```bash (package)
```

```bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's narwhal primary address.

```bash (package)
```

```bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's narwhal worker address.

```bash (package)
```

```bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash (package)
```

```bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of worker key.

```bash (package)
```

```bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of network key.

```bash (package)
```

```bash
bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
bash (package)
```

```bash
bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
bash (package)
```

```bash
bash (package)
```

Returns unix timestamp of the start of current epoch

```bash
bash (package)
```

```bash
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
bash (package)
```

```bash
bash (package)
```

Returns all the validators who are currently reporting addr

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

Function

Update a validator's name.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's description

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's image url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's project url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's network address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash
```

```bash
```

Update candidate validator's p2p address.

```bash
```

```bash
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash
```

```bash
```

Update candidate validator's narwhal primary address.

```bash
```

```bash
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash
```

```bash
```

Update candidate validator's narwhal worker address.

```bash
```

```bash
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
```

```bash
```

Update candidate validator's public key of protocol key and proof of possession.

```bash
```

```bash
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
```

```bash
```

Update candidate validator's public key of worker key.

```bash
```

```bash
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
```

```bash
```

Update candidate validator's public key of network key.

```bash
```

```bash
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash (package)```

```bash (package)```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash (package)```

```bash (package)```

Returns unix timestamp of the start of current epoch

```bash (package)```

```bash (package)```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)```

```bash (package)```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)```

```bash (package)```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash (package)```

```bash (package)```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash (package)```

```bash (package)```

Returns all the validators who are currently reporting addr

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

Function

Update a validator's description

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's image url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's project url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's network address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's p2p address.

```bash (package)
```

```bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's narwhal primary address.

```bash (package)
```

```bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's narwhal worker address.

```bash (package)
```

```bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash (package)
```

```bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of worker key.

```bash (package)
```

```bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of network key.

```bash (package)
```

```bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash (package)
```

```bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

`bash (package)`

`bash (package)`

Returns unix timestamp of the start of current epoch

`bash (package)`

`bash (package)`

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

`bash (package)`

`bash (package)`

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

`bash (package)`

`bash (package)`

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

`bash (package)`

`bash (package)`

Returns reference to the staking pool mappings that map pool ids to active validator addresses

`bash (package)`

`bash (package)`

Returns all the validators who are currently reporting addr

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

```bash
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

Function

Update a validator's image url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's project url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's network address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's p2p address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal primary address.

```bash
bash (package)
```

```bash
(package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's narwhal worker address.

```bash
(package)
```

```bash
(package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash
(package)
```

```bash
(package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of worker key.

```bash
(package)
```

```bash
(package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of network key.

```bash
(package)
```

```bash
(package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
(package)
```

```bash
(package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never

change for a given network.

```bash
(package)
```

```bash
(package)
```

Returns unix timestamp of the start of current epoch

```bash
(package)
```

```bash
(package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
(package)
```

```bash
(package)
```

Returns all the validators who are currently reporting addr

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash

```

```bash

```

```bash
(package)
```

```bash
bash (package)
```

```bash

```

```bash

```

Function

Update a validator's project url

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's network address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's p2p address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal primary address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal worker address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```
bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```
bash (package)
```

```
bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's public key of worker key.

```
bash (package)
```

```
bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's public key of network key.

```
bash (package)
```

```
bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```
bash (package)
```

```
bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```
bash (package)
```

```
bash (package)
```

Returns unix timestamp of the start of current epoch

```
bash (package)
```

```
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```
bash (package)
```

```
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
(package)
```

```bash
(package)
```

Returns all the validators who are currently reporting addr

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

```bash
```

```bash
```

Function

Update a validator's network address. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's network address.

```bash (package)
```

```bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's p2p address.

```bash (package)
```

```bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's narwhal primary address.

```bash (package)
```

```bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's narwhal worker address.

```bash (package)
```

```bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash (package)
```

```bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of worker key.

```bash (package)
```

```bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of network key.

```bash (package)
```

```bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash (package)
```

```bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash (package)
```

```bash (package)
```

Returns unix timestamp of the start of current epoch

```bash (package)
```

```bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)
```

```bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)
```

```bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash (package)
```

```bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash (package)
```

```bash (package)
```

Returns all the validators who are currently reporting addr

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

Function

Update candidate validator's network address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's p2p address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal primary address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's narwhal worker address.

```bash (package)
```

```bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash (package)
```

```bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of worker key.

```bash (package)
```

```bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of network key.

```bash (package)
```

```bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash (package)
```

```bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash (package)
```

```bash
(package)
```

Returns unix timestamp of the start of current epoch

```bash
(package)
```

```bash
(package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
(package)
```

```bash
(package)
```

Returns all the validators who are currently reporting addr

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

```bash
```

```

```bash

```

Function

Update a validator's p2p address. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's p2p address.

```
bash (package)
```

```
bash (package)
```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's narwhal primary address.

```
bash (package)
```

```
bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's narwhal worker address.

```
bash (package)
```

```
bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```
bash (package)
```

```
bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's public key of worker key.

```
bash (package)
```

```
bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

`bash (package)`

`bash (package)`

Update candidate validator's public key of network key.

`bash (package)`

`bash (package)`

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

`bash (package)`

`bash (package)`

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

`bash (package)`

`bash (package)`

Returns unix timestamp of the start of current epoch

`bash (package)`

`bash (package)`

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

`bash (package)`

`bash (package)`

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

`bash (package)`

`bash (package)`

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

`bash (package)`

`bash (package)`

Returns reference to the staking pool mappings that map pool ids to active validator addresses

`bash (package)`

`bash (package)`

Returns all the validators who are currently reporting addr

`bash (package)`

`bash (package)`

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash (package)```

```bash (package)```

```bash
```

```bash
```

Function

Update candidate validator's p2p address.

```bash (package)```

```bash (package)```

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash (package)```

```bash (package)```

Update candidate validator's narwhal primary address.

```bash (package)```

```bash (package)```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash (package)```

```bash (package)```

Update candidate validator's narwhal worker address.

```bash (package)```

```bash
(package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash
(package)
```

```bash
(package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of worker key.

```bash
(package)
```

```bash
(package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of network key.

```bash
(package)
```

```bash
(package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
(package)
```

```bash
(package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
(package)
```

```bash
(package)
```

Returns unix timestamp of the start of current epoch

```bash
(package)
```

```bash
(package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)```

```bash (package)```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)```

```bash (package)```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash (package)```

```bash (package)```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash (package)```

```bash (package)```

Returns all the validators who are currently reporting addr

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

Extract required Balance from vector of Coin, transfer the remainder back to sender.

```bash
```

```bash
```

```bash (package)```

```bash (package)```

```bash
```

```bash
```

Function

Update a validator's narwhal primary address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's narwhal primary address.

```bash (package)
```

```bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's narwhal worker address.

```bash (package)
```

```bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash (package)
```

```bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of worker key.

```bash (package)
```

```bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of network key.

```bash (package)
```

```bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash (package)
```

```bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
bash (package)
```

```bash
bash (package)
```

Returns unix timestamp of the start of current epoch

```bash
bash (package)
```

```bash
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
bash (package)
```

```bash
bash (package)
```

Returns all the validators who are currently reporting addr

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

Function

Update candidate validator's narwhal primary address.

```
bash (package)
```

```
bash (package)
```

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's narwhal worker address.

```
bash (package)
```

```
bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```
bash (package)
```

```
bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's public key of worker key.

```
bash (package)
```

```
bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```
bash (package)
```

```bash (package)
```

Update candidate validator's public key of network key.

```bash (package)
```

```bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash (package)
```

```bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash (package)
```

```bash (package)
```

Returns unix timestamp of the start of current epoch

```bash (package)
```

```bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)
```

```bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)
```

```bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash (package)
```

```bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash (package)
```

```bash (package)
```

Returns all the validators who are currently reporting addr

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

Function

Update a validator's narwhal worker address. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's narwhal worker address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
(package)
```

Update candidate validator's public key of worker key.

```bash
(package)
```

```bash
(package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of network key.

```bash
(package)
```

```bash
(package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
(package)
```

```bash
(package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
(package)
```

```bash
(package)
```

Returns unix timestamp of the start of current epoch

```bash
(package)
```

```bash
(package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
bash (package)
```

```bash
bash (package)
```

Returns all the validators who are currently reporting addr

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash

```

```bash

```

```bash
bash (package)
```

```bash
bash (package)
```

```bash

```

```bash

```

Function

Update candidate validator's narwhal worker address.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash
bash (package)
```

```bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of worker key.

```bash (package)
```

```bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of network key.

```bash (package)
```

```bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash (package)
```

```bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash (package)
```

```bash (package)
```

Returns unix timestamp of the start of current epoch

```bash (package)
```

```bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)
```

```bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)
```

```bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
(package)
```

```bash
(package)
```

Returns all the validators who are currently reporting addr

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Extract required Balance from vector of Coin, transfer the remainder back to sender.

```bash

```

```bash

```

```bash
(package)
```

```bash
(package)
```

```bash

```

```bash

```

Function

Update a validator's public key of protocol key and proof of possession. The change will only take effects starting from the next epoch.

```bash
(package)
```

```bash
(package)
```

Update candidate validator's public key of protocol key and proof of possession.

```bash
(package)
```

```
bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's public key of worker key.

```
bash (package)
```

```
bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's public key of network key.

```
bash (package)
```

```
bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```
bash (package)
```

```
bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```
bash (package)
```

```
bash (package)
```

Returns unix timestamp of the start of current epoch

```
bash (package)
```

```
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```
bash (package)
```

```
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```
bash (package)
```

```
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
bash (package)
```

```bash
bash (package)
```

Returns all the validators who are currently reporting addr

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

Function

Update candidate validator's public key of protocol key and proof of possession.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of worker key.

```bash (package)
```

```bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash (package)
```

```bash (package)
```

Update candidate validator's public key of network key.

```bash (package)
```

```bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash (package)
```

```bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

```bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash (package)
```

```bash (package)
```

Returns unix timestamp of the start of current epoch

```bash (package)
```

```bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)
```

```bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)
```

```bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash (package)
```

```bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash (package)
```

```bash
bash (package)
```

Returns all the validators who are currently reporting addr

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

Function

Update a validator's public key of worker key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of worker key.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of network key.

```bash
bash (package)
```

```bash
bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
bash (package)
```

```bash
bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
bash (package)
```

```bash
bash (package)
```

Returns unix timestamp of the start of current epoch

```bash
bash (package)
```

```bash
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
bash (package)
```

```bash
bash (package)
```

Returns all the validators who are currently reporting addr

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

Function

Update candidate validator's public key of worker key.

```bash
bash (package)
```

```bash
bash (package)
```

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```bash
bash (package)
```

```bash
bash (package)
```

Update candidate validator's public key of network key.

```bash
bash (package)
```

```bash
bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
bash (package)
```

```bash
bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
bash (package)
```

```bash
bash (package)
```

Returns unix timestamp of the start of current epoch

```bash
bash (package)
```

```bash
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
bash (package)
```

```bash
bash (package)
```

Returns all the validators who are currently reporting addr

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin, transfer the remainder back to sender.

```bash
```

```

```bash

```

```
bash (package)
```

```
bash (package)
```

```bash

```

```bash

```

Function

Update a validator's public key of network key. The change will only take effects starting from the next epoch.

```
bash (package)
```

```
bash (package)
```

Update candidate validator's public key of network key.

```
bash (package)
```

```
bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```
bash (package)
```

```
bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```
bash (package)
```

```
bash (package)
```

Returns unix timestamp of the start of current epoch

```
bash (package)
```

```
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```
bash (package)
```

```
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

`bash (package)`

`bash (package)`

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

`bash (package)`

`bash (package)`

Returns reference to the staking pool mappings that map pool ids to active validator addresses

`bash (package)`

`bash (package)`

Returns all the validators who are currently reporting addr

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

`bash (package)`

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash

```

```bash

```

`bash (package)`

`bash (package)`

```bash

```

```bash

```

Function

Update candidate validator's public key of network key.

`bash (package)`

```bash
bash (package)
```

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
bash (package)
```

```bash
bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
bash (package)
```

```bash
bash (package)
```

Returns unix timestamp of the start of current epoch

```bash
bash (package)
```

```bash
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
bash (package)
```

```bash
bash (package)
```

Returns all the validators who are currently reporting addr

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash

```

```bash

```

```bash
bash (package)
```

```bash
bash (package)
```

```bash

```

```bash

```

Function

This function should be called at the end of an epoch, and advances the system to the next epoch. It does the following things:

```bash
bash (package)
```

```bash
bash (package)
```

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
bash (package)
```

```bash
bash (package)
```

Returns unix timestamp of the start of current epoch

```bash
bash (package)
```

```bash
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
bash (package)
```

```bash
bash (package)
```

Returns all the validators who are currently reporting addr

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

Function

Return the current epoch number. Useful for applications that need a coarse-grained concept of time, since epochs are ever-increasing and epoch changes are intended to happen every 24 hours.

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash (package)```

```bash (package)```
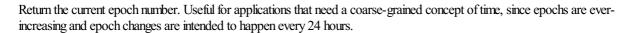
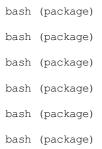Returns unix timestamp of the start of current epoch

```bash (package)```

```bash (package)```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)```

```bash (package)```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash (package)```

```bash (package)```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash (package)```

```bash (package)```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash (package)```

```bash (package)```

Returns all the validators who are currently reporting addr

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

```bash (package)```

bash (package)

bash (package)

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash

```

```bash

```

bash (package)

bash (package)

```bash

```

```bash

```

Function

bash (package)

bash (package)

bash (package)

bash (package)

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

bash (package)

bash (package)

Returns unix timestamp of the start of current epoch

bash (package)

bash (package)

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

bash (package)

bash (package)

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

bash (package)

bash (package)

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

bash (package)

bash (package)

Returns reference to the staking pool mappings that map pool ids to active validator addresses

bash (package)

```bash
(package)
```

Returns all the validators who are currently reporting addr

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
(package)
```

```bash
(package)
```

```bash
```

```bash
```

```bash
```

```bash
```

Function

```bash
(package)
```

```bash
(package)
```

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
(package)
```

```bash
(package)
```

Returns unix timestamp of the start of current epoch

```bash
(package)
```

```bash
(package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
bash (package)
```

```bash
bash (package)
```

Returns all the validators who are currently reporting addr

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin, transfer the remainder back to sender.

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

Function

This function always return the genesis system state version, which is used to create the system state in genesis. It should never change for a given network.

```bash
(package)
```

```bash
(package)
```

Returns unix timestamp of the start of current epoch

```bash
(package)
```

```bash
(package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
(package)
```

```bash
(package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
(package)
```

```bash
(package)
```

Returns all the validators who are currently reporting addr

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

```bash
(package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
```

```bash
```

Function

Returns unix timestamp of the start of current epoch

```bash
bash (package)
```

```bash
bash (package)
```

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```bash
bash (package)
```

```bash
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```bash
bash (package)
```

```bash
bash (package)
```

Returns all the validators who are currently reporting addr

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

```bash
bash (package)
```

Extract required Balance from vector of Coin, transfer the remainder back to sender.

```bash
```

```bash
```

bash (package)

bash (package)

```bash
```

```bash
```

Function

Returns the total amount staked with validator_addr. Aborts if validator_addr is not an active validator.

bash (package)

bash (package)

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

bash (package)

bash (package)

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

bash (package)

bash (package)

Returns reference to the staking pool mappings that map pool ids to active validator addresses

bash (package)

bash (package)

Returns all the validators who are currently reporting addr

bash (package)

bash (package)

bash (package)

bash (package)

bash (package)

bash (package)

bash (package)

bash (package)

bash (package)

bash (package)

bash (package)

bash (package)

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

Function

Returns the voting power for validator_addr. Aborts if validator_addr is not an active validator.

```
bash (package)
```

```
bash (package)
```

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.
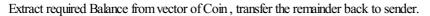
```
bash (package)
```

```
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```
bash (package)
```

```
bash (package)
```

Returns all the validators who are currently reporting addr

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```
```

```bash
```

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

Function

Returns the staking pool id of a given validator. Aborts if validator_addr is not an active validator.

```
bash (package)
```

```
bash (package)
```

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```
bash (package)
```

```
bash (package)
```

Returns all the validators who are currently reporting addr

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

Function

Returns reference to the staking pool mappings that map pool ids to active validator addresses

```
bash (package)
```

```
bash (package)
```

Returns all the validators who are currently reporting addr

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

Function

Returns all the validators who are currently reporting addr

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

Function

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

bash (package)

bash (package)

```bash
```

```bash
```

Function

bash (package)

bash (package)

bash (package)

bash (package)

bash (package)

bash (package)

bash (package)

bash (package)

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

bash (package)

bash (package)

```bash
```

```bash
```

Function

bash (package)

bash (package)

bash (package)

bash (package)

bash (package)

bash (package)

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

Function

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

```
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```
bash (package)
```

```
bash (package)
```

```bash
```

```bash
```

Function

```
bash (package)
```

```
bash (package)
```

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

```
```

bash (package)

bash (package)

```bash
```

```bash
```

Function

Extract required Balance from vector of Coin , transfer the remainder back to sender.

```bash
```

```bash
```

bash (package)

bash (package)

```bash
```

```bash
```

Function

bash (package)

bash (package)

```bash
```

```bash
```

Macro function

```bash
```

```bash
```