

# Module sui::pay

This module provides handy functionality for wallets and sui::Coin management.

For when empty vector is supplied into join function.

Transfer c to the sender of the current transaction

Split coin self to two coins, one with balance split\_amount, and the remaining balance is left is self.

Split coin self into multiple coins, each with balance specified in split\_amounts. Remaining balance is left in self.

Send amount units of c to recipient Aborts with EVALUE if amount is greater than or equal to amount

Divide coin self into n - 1 coins with equal balances. If the balance is not evenly divisible by n, the remainder is left in self.

Join [coin](#) into self. Re-exports [coin::join](#) function. Deprecated: you should call [coin](#) . [join](#) (other) directly.

Join everything in coins with self

Join a vector of Coin into a single object and transfer it to receiver.

## Constants

For when empty vector is supplied into join function.

```
```bash
```

```
```
```

Transfer c to the sender of the current transaction

```
```bash
```

```
```
```

```
```bash
```

```
```
```

Split coin self to two coins, one with balance split\_amount, and the remaining balance is left is self.

```
```bash
```

```
```
```

```
```bash
```

```
```
```

Split coin self into multiple coins, each with balance specified in split\_amounts. Remaining balance is left in self.

```
```bash
```

```
```
```

```
```bash
```

```
```
```

Send amount units of c to recipient Aborts with EVALUE if amount is greater than or equal to amount

```
```bash
```

```
```
```

```
'''bash
```

```
'''
```

Divide coin self into  $n - 1$  coins with equal balances. If the balance is not evenly divisible by  $n$ , the remainder is left in self.

```
'''bash
```

```
'''
```

```
'''bash
```

```
'''
```

Join [coin](#) into self. Re-exports [coin.join](#) function. Deprecated: you should call [coin.join](#) (other) directly.

```
'''bash
```

```
'''
```

```
'''bash
```

```
'''
```

Join everything in coins with self

```
'''bash
```

```
'''
```

```
'''bash
```

```
'''
```

Join a vector of Coin into a single object and transfer it to receiver.

```
'''bash
```

```
'''
```

```
'''bash
```

```
'''
```

## Function

Transfer  $c$  to the sender of the current transaction

```
'''bash
```

```
'''
```

```
'''bash
```

```
'''
```

Split coin self to two coins, one with balance `split_amount`, and the remaining balance is left in self.

```
'''bash
```

```
'''
```

```
'''bash
```

```
'''
```

Split coin self into multiple coins, each with balance specified in `split_amounts`. Remaining balance is left in self.

```
```bash
```

```
```
```

```
```bash
```

```
```
```

Send amount units of c to recipient Aborts with EVALUE if amount is greater than or equal to amount

```
```bash
```

```
```
```

```
```bash
```

```
```
```

Divide coin self into n - 1 coins with equal balances. If the balance is not evenly divisible by n, the remainder is left in self.

```
```bash
```

```
```
```

```
```bash
```

```
```
```

Join [coin](#) into self. Re-exports [coin::join](#) function. Deprecated: you should call [coin . join](#) (other) directly.

```
```bash
```

```
```
```

```
```bash
```

```
```
```

Join everything in coins with self

```
```bash
```

```
```
```

```
```bash
```

```
```
```

Join a vector of Coin into a single object and transfer it to receiver.

```
```bash
```

```
```
```

```
```bash
```

```
```
```

## Function

Split coin self to two coins, one with balance split\_amount, and the remaining balance is left is self.

```
```bash
```

```
```
```

```
```bash
```

```
'''
```

Split coin self into multiple coins, each with balance specified in split\_amounts. Remaining balance is left in self.

```
'''bash
```

```
'''
```

```
'''bash
```

```
'''
```

Send amount units of c to recipient Aborts with EVALUATE if amount is greater than or equal to amount

```
'''bash
```

```
'''
```

```
'''bash
```

```
'''
```

Divide coin self into n - 1 coins with equal balances. If the balance is not evenly divisible by n, the remainder is left in self.

```
'''bash
```

```
'''
```

```
'''bash
```

```
'''
```

Join [coin](#) into self. Re-exports [coin::join](#) function. Deprecated: you should call [coin](#) . [join](#) (other) directly.

```
'''bash
```

```
'''
```

```
'''bash
```

```
'''
```

Join everything in coins with self

```
'''bash
```

```
'''
```

```
'''bash
```

```
'''
```

Join a vector of Coin into a single object and transfer it to receiver.

```
'''bash
```

```
'''
```

```
'''bash
```

```
'''
```

## Function

Split coin self into multiple coins, each with balance specified in split\_amounts. Remaining balance is left in self.

```
'''bash
```

```
'''
```

```
'''bash
```

```
'''
```

Send amount units of c to recipient Aborts with EVALUE if amount is greater than or equal to amount

```
'''bash
```

```
'''
```

```
'''bash
```

```
'''
```

Divide coin self into n - 1 coins with equal balances. If the balance is not evenly divisible by n, the remainder is left in self.

```
'''bash
```

```
'''
```

```
'''bash
```

```
'''
```

Join [coin](#) into self. Re-exports [coin:join](#) function. Deprecated: you should call [coin](#) . [join](#) (other) directly.

```
'''bash
```

```
'''
```

```
'''bash
```

```
'''
```

Join everything in coins with self

```
'''bash
```

```
'''
```

```
'''bash
```

```
'''
```

Join a vector of Coin into a single object and transfer it to receiver.

```
'''bash
```

```
'''
```

```
'''bash
```

```
'''
```

## Function

Send amount units of c to recipient Aborts with EVALUE if amount is greater than or equal to amount

```
'''bash
```

```
'''
```

```
'''bash
```

```
'''
```

Divide coin self into  $n - 1$  coins with equal balances. If the balance is not evenly divisible by  $n$ , the remainder is left in self.

```
```bash
```

```
```
```

```
```bash
```

```
```
```

Join [coin](#) into self. Re-exports [coin::join](#) function. Deprecated: you should call [coin](#) . [join](#) (other) directly.

```
```bash
```

```
```
```

```
```bash
```

```
```
```

Join everything in coins with self

```
```bash
```

```
```
```

```
```bash
```

```
```
```

Join a vector of Coin into a single object and transfer it to receiver.

```
```bash
```

```
```
```

```
```bash
```

```
```
```

## Function

Divide coin self into  $n - 1$  coins with equal balances. If the balance is not evenly divisible by  $n$ , the remainder is left in self.

```
```bash
```

```
```
```

```
```bash
```

```
```
```

Join [coin](#) into self. Re-exports [coin::join](#) function. Deprecated: you should call [coin](#) . [join](#) (other) directly.

```
```bash
```

```
```
```

```
```bash
```

```
```
```

Join everything in coins with self

```
```bash
```

```
```
```

```
```bash
```

```
```
```

Join a vector of Coin into a single object and transfer it to receiver.

```
```bash
```

```
```
```

```
```bash
```

```
```
```

## Function

Join [coin](#) into self. Re-exports [coin::join](#) function. Deprecated: you should call [coin](#) . [join](#) (other) directly.

```
```bash
```

```
```
```

```
```bash
```

```
```
```

Join everything in coins with self

```
```bash
```

```
```
```

```
```bash
```

```
```
```

Join a vector of Coin into a single object and transfer it to receiver.

```
```bash
```

```
```
```

```
```bash
```

```
```
```

## Function

Join everything in coins with self

```
```bash
```

```
```
```

```
```bash
```

```
```
```

Join a vector of Coin into a single object and transfer it to receiver.

```
```bash
```

```
```
```

```
```bash
```

'''

## Function

Join a vector of Coin into a single object and transfer it to receiver.

'''bash

'''

'''bash

'''