

交易的Hash

两个最常用的自定义数据类型**common.Hash**用来表示哈希值，**common.Address**表示地址

```
# /commons/types.go
const (
    HashLength = 32
    AddressLength = 20
)
type Hash [HashLength]byte
type Address [AddressLength]byte
```

在Ethereum 代码里，所有用到的哈希值，都使用该Hash类型，长度为32bytes，即256 bits；Ethereum 中所有跟帐号(Account)相关的信息，比如交易转帐的转出帐号(地址)和转入帐号(地址)，都会用该Address类型表示，长度20bytes。

big.Int是golang提供的数据类型，用来处理比较大的整型数，当然它也可以处理诸如64bit, 32bit的常用整数。

```
# /go-1.x/src/math/big/int.go
package big
type Int struct {
    neg bool    // sign, whether negative
    abs nat     // absolute value of integer
}
```

big.Int是一个结构体(struct)，相当于C++中的class，所以每次新建big.Int时可以用 `x := new(big.Int)`，返回一个指针。注意对Int的算术操作，要使用该对象的成员函数，比如Add()：

```
func (z *Int) Add(x, y *Int) *Int    // Add sets z to sum x+y and returns z
```