



```
#region DTOs
6 references
public class TransactionDetail
{
    3 references
    public int TransactionNumber { get; set; }
    3 references
    public string AccountNumber { get; set; }
    3 references
    public double Amount { get; set; }
}

5 references
public class Account
{
    2 references
    public string AccountNumber { get; set; }
    2 references
    public List<Transaction> Transactions { get; set; }
}

6 references
public class Transaction
{
    3 references
    public int TransactionNumber { get; set; }
    3 references
    public double Amount { get; set; }
}

#endregion

0 references
private List<Account> ExtractAccounts(List<TransactionDetail> transactionDetails)
{

```

```

[TestClass]
0 references
public class UnitTest
{
    [TestMethod]
    0 references
    public void TestExtract()
    {
        List<TransactionDetail> transactionDetailsInput = new List<TransactionDetail>()
        {
            new TransactionDetail()
            {
                AccountNumber = "0001123456",
                TransactionNumber = 1,
                Amount = 10000
            },
            new TransactionDetail()
            {
                AccountNumber = "WRAP9876564",
                TransactionNumber = 10,
                Amount = 30000
            },
            new TransactionDetail()
            {
                AccountNumber = "0001123456",
                TransactionNumber = 20,
                Amount = 200
            }
        };

        List<Account> expectedAccounts = new List<Account>()
        {
            new Account()
            {
                AccountNumber = "0001123456",
                Transactions = new List<Transaction>()
                {
                    new Transaction()
                    {
                        TransactionNumber = 1,
                        Amount = 10000
                    },
                    new Transaction()
                    {
                        TransactionNumber = 20,
                        Amount = 200
                    }
                }
            },
            new Account()
            {
                AccountNumber = "WRAP9876564",
                Transactions = new List<Transaction>()
                {
                    new Transaction()
                    {
                        TransactionNumber = 10,
                        Amount = 30000
                    }
                }
            }
        };

        Assert.AreEqual(expectedAccounts, ExtractAccounts(transactionDetailsInput));
    }
}

```