

```
using System;
using System.Collections.Generic;
using System.Text;
using System.Linq;

using Microsoft.Modeling;

namespace TypeBindingSample
{
    public static class Model
    {
        static SetContainer<AccountDefinition> accountSet = new SetContainer<AccountDefinition>();

        //remember no condition -> can always fire
        [Rule]
        static AccountDefinition CreateAccount()
        {
            AccountDefinition account = new AccountDefinition();
            accountSet.Add(account);
            return account;
        }

        [Rule]
        static void SetBalance(AccountDefinition account, float balance)
        {
            account.Balance = balance;
        }

        [Rule]
        static float GetBalance(AccountDefinition account)
        {
            return account.Balance;
        }

        [Rule]
        static Set<AccountDefinition> SearchAccounts(float balance)
        {
            SetContainer<AccountDefinition> set = new SetContainer<AccountDefinition>();
            var matchingAccounts = from account in accountSet where account.Balance == balance select
account;
            foreach (AccountDefinition account in matchingAccounts)
            {
                set.Add(account);
            }
            return set.ToSet();
        }

        [Rule]
        static void Clear()
        {
            accountSet.Clear();
        }
    }

    /// <summary>
    /// Model type AccountDefinition is bound to implementation type Account
    /// </summary>
    [TypeBinding("TypeBindingSample.Account")]
    public class AccountDefinition
    {
        public float Balance { get; set; }
    }
}
```

