

```
1
2 namespace Wk6InheritanceExample4
3 {
4     interface IPayDetail
5     {
6         string GetPayDetailInformation();
7     }
8 }
9
```

```
1
2 namespace Wk6InheritanceExample4
3 {
4     class Person
5     {
6         private string _Name;
7
8         public string Name { get { return _Name; } }
9
10        public Person(string name)
11        {
12            _Name = name;
13        }
14
15        public override string ToString()
16        {
17            return _Name;
18        }
19    }
20 }
```

```
1
2
3 namespace Wk6InheritanceExample4
4 {
5     abstract class Employee : IPayDetail
6     {
7         private Person _ContactDetails;
8         protected decimal _Amount;
9
10        public Person ContactDetails
11        { get { return _ContactDetails; } }
12
13        public abstract string GetPayDetailInformation();
14
15
16        public Employee(string name, decimal amount)
17        {
18            _ContactDetails = new Person(name);
19            _Amount = amount;
20        }
21
22        public override string ToString()
23        {
24            return string.Format("{0}", _ContactDetails);
25        }
26    }
27 }
28
```

```
1
2
3 namespace Wk6InheritanceExample4
4 {
5     class Salary : Employee
6     {
7         public decimal AnnualSalary
8         {
9             get { return _Amount; }
10            set { _Amount = value; }
11        }
12
13        public override string GetPayDetailInformation()
14        {
15            return string.Format("{0:c} salary - Weekly Payment: {1:c}",
16                                  _Amount, _Amount/52);
17        }
18
19        public Salary(string name, decimal amount): base(name, amount)
20        { }
21
22        public override string ToString()
23        {
24            return string.Format("{0} - {1}", base.ToString(),
25                                  GetPayDetailInformation());
26        }
27    }
28 }
```

```
1
2
3 namespace Wk6InheritanceExample4
4 {
5     class Wage : Employee
6     {
7
8         public decimal Rate { get { return _Amount; } set { _Amount =
9             value; } }
10
11         public Wage(string name, decimal rate) : base(name, rate)
12         {
13
14             public override string GetPayDetailInformation()
15             {
16                 return string.Format("{0:c} Hour Rate - Weekly Payment: {1:c}",
17                     _Amount, _Amount * 36 );
18             }
19
20             public override string ToString()
21             {
22                 return string.Format("{0} - {1}", base.ToString(),
23                     GetPayDetailInformation());
24             }
25         }
26     }
27 }
```

```
1 using System.Collections.Generic;
2 using System.Collections.ObjectModel;
3
4 namespace Wk6InheritanceExample4
5 {
6     class Company
7     {
8         private string _CompanyName;
9         private List<Employee> _Employees;
10
11         public string CompanyName
12         { get { return _CompanyName; } }
13
14         public ReadOnlyCollection<Employee> Employees
15         { get { return _Employees.AsReadOnly(); } }
16
17         public Company(string name)
18         {
19             _CompanyName = name;
20             _Employees = new List<Employee>();
21         }
22
23         public Company(Company acompany)
24         {
25             _CompanyName = acompany.CompanyName;
26             _Employees = new List<Employee>();
27             foreach (Employee e in acompany.Employees)
28                 _Employees.Add(e); // use shallow copy here in the case the ↗
                                     employee change in other branch it will share same object
29         }
30
31         ~Company()
32         {
33             if (_Employees.Count != 0) _Employees.Clear();
34         }
35
36         public void Add(Employee e) { _Employees.Add(e); }
37
38         public static Company operator+(Company c, Employee e)
39         {
40             Company temp = new Company(c);
41             temp.Add(e);
42             return temp;
43         }
44
45         public override string ToString()
46         {
47             string temp = string.Format("{0}\n", _CompanyName);
48             foreach (Employee e in _Employees) temp += e + "\n";
49             return temp;
50         }
51     }
52 }
```

```
1 using System;
2 using System.Collections.Generic;
3
4 namespace Wk6InheritanceExample4
5 {
6     class Program
7     {
8         static void Main(string[] args)
9         {
10             Company abc = new Company("ABC Rental Company");
11
12             abc += (new Salary("Benita", 60000m));
13             abc += (new Wage("Travis", 25m));
14
15
16             Console.WriteLine(abc);
17
18         }
19     }
20 }
21
```