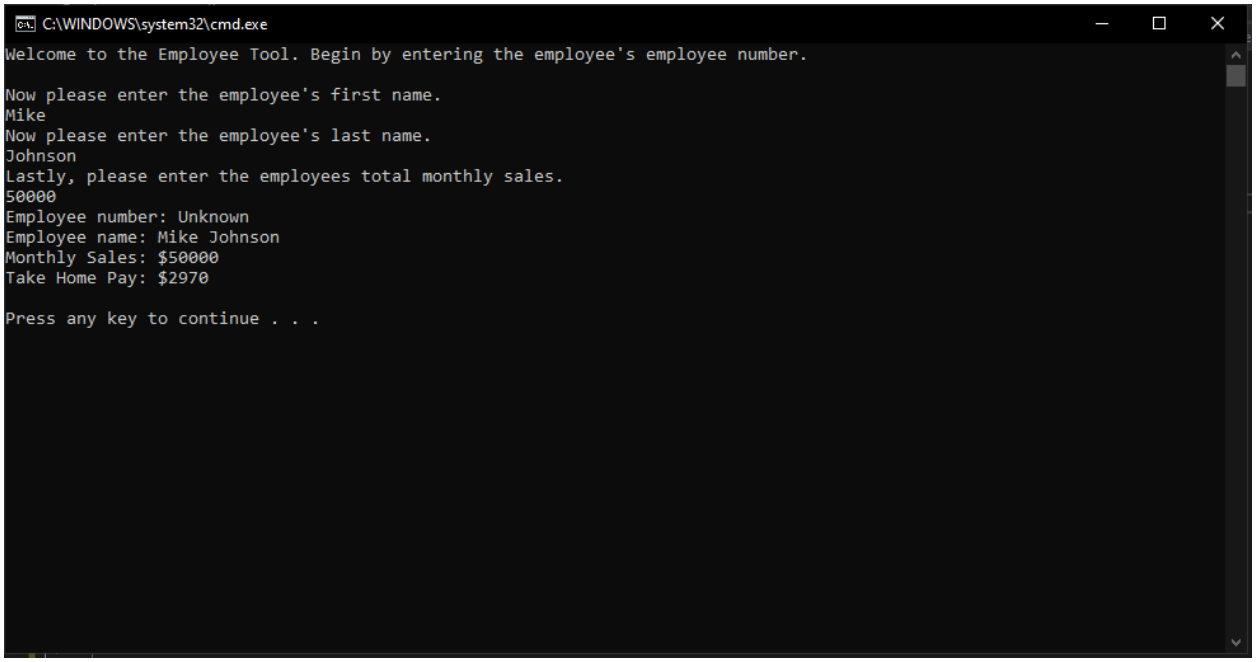

Name
-firstName: string -lastName: string
+Name() +Name(string,string) +Name(string firstname) +DisplayName(): void +SetLastName (string): void +SetFirstName (string): void +GetFirstName (string): void +GetLastName (string): void +FirstName: string +LastName: string +ToString(): string ~Name{}

Has-a Name

Employee
-employeeNum: string -employeeName: Name -monthlySalary: double
+Employee() +Employee(string, Name, double) +Employee(string, string double)

```
+Employee(string, double)
+Employee(string, string, string, double)
+EmployeeName: Name
+TakeHome(double): double
+ToString(): string
```

```
namespace HW_9
{
   class Name
   {
      private string firstName;
      private string lastName;
      public Name()
      {
         firstName = "Unknown";
         lastName = "Unknown";
      }
      public Name(string fn, string ln)
      {
         firstName = fn;
         lastName = ln;
      }
      public Name(string fn)
      {
         firstName = fn;
         lastName = "Unknown";
      }
      public void DisplayName()
      {
         Console.WriteLine("The name is: {0} {1}", firstName, lastName);
      }
      //mutators
      public void SetLastName(string newLN)
      {
         lastName = newLN;
      }
      public void SetFirstName(string newFN)
      {
         firstName = newFN;
      }
      //Accessors
      public string GetFirstName()
      {
         return firstName;
```

```
        }
        public string GetLastName()
        {
            return lastName;
        }
        //properties
        public string FirstName
        {
            get { return firstName; }
            set { firstName = value; }
        }
        public string LastName
        {
            get { return lastName; }
            set { lastName = value; }
        }
        public override string ToString()
        {
            return firstName + " " + lastName;
        }

        //destructor
        ~Name() { }
    }
}


namespace HW_9
{
    class Employee
    {
        private string employeeNum;
        private Name employeeName;
        private double monthlySales;

        public Employee()
        {
            employeeNum = "123456";
            employeeName = new Name();
            monthlySales = 10000;
        }

        public Employee(string num, Name nm, double sal)
        {
```

```
    employeeNum = num;
    employeeName = nm;
    monthlySales = sal;
}

public Employee(string fn, string ln, double sal)
{
    employeeNum = "Unknown";
    employeeName = new Name(fn, ln);
    monthlySales = sal;
}

public Employee(string eNum, double sal)
{
    employeeNum = eNum;
    employeeName = new Name();
    monthlySales = sal;
}

public Employee(string num, string fn, string ln, double sal)
{
    employeeNum = num;
    employeeName = new Name(fn, ln);
    monthlySales = sal;
}

public Name EmployeeName
{
    get { return employeeName; }
    set { employeeName = value; }
}

public double TakeHome(double monthlySale)
{
    double taxedSal = monthlySale * 0.09;
    double actualMoney = taxedSal - (taxedSal * 0.18) - (taxedSal * 0.1) - (taxedSal * 0.06);
    actualMoney = Math.Round(actualMoney, 2);
    return actualMoney;
}

public override string ToString()
{
    string outStr = "Employee number: " + employeeNum + "\n";
    outStr += "Employee name: " + employeeName.ToString() + "\n";
```

```
            outStr += "Monthly Sales: $" + monthlySales.ToString() + "\n";
            outStr += "Take Home Pay: $" + TakeHome(monthlySales).ToString() + "\n";
            return outStr;
        }
    }
}


namespace HW_9
{
    class Program
    {

        static void Main(string[] args)
        {
            double monthlySales;
            Employee[] emp = new Employee[1];

            Console.WriteLine("Welcome to the Employee Tool. Begin by entering the employee's
employee number.");
            string eNum = Console.ReadLine();
            Console.WriteLine("Now please enter the employee's first name.");
            string fn = Console.ReadLine();
            Console.WriteLine("Now please enter the employee's last name.");
            string ln = Console.ReadLine();
            Console.WriteLine("Lastly, please enter the employees total monthly sales.");
            while (true)
            {
                string monthSale = Console.ReadLine();

                if (double.TryParse(monthSale, out monthlySales)&&monthlySales>=0)
                {

                    if (String.IsNullOrEmpty(eNum))
                    {
                        emp[0] = new Employee(fn, ln, monthlySales);
                        Console.WriteLine(emp[0].ToString());
                        break;
                    }

                    else if (String.IsNullOrEmpty(fn)&String.IsNullOrEmpty(ln))
                    {
                        emp[0] = new Employee(eNum, monthlySales);
                        Console.WriteLine(emp[0].ToString());
```

```
                    break;
                }

                else
                {
                    emp[0] = new Employee(eNum, fn, ln, monthlySales);
                    Console.WriteLine(emp[0].ToString());
                    break;
                }

            }
            else
            {
                Console.WriteLine("Invalid input, please re-enter");
            }
        }


    }
  }
}
```