

بسم الله الرحمن الرحيم

آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

قالب بندی نمایش داده ها

مدرس : مهندس افشین رفوآ

قالب بندی نمایش داده ها

1. برای این منظور، فایل را به ترتیب زیر تغییر دهید.

```
using System;
public class OrderProcessing
{
public static int Main()
    {
        // Price of items
        const double PriceOneShirt = 0.95;
        const double PriceAPairOfPants = 2.95;
        const double PriceOneDress = 4.55;
        const double TaxRate = 0.0575; // 5.75%
        // Basic information about an order
        homePhone; string customerName
        DateTime orderDate;
        // Unsigned numbers to represent cleaning items
        numberOfDresses; numberOfPants; uint numberOfShirts
        // Each of these sub totals will be used for cleaning items
        subTotalDresses; subTotalPants; double subTotalShirts
        // Values used to process an order
        salesTotal; taxAmount; double totalOrder
        moneyChange; double amountTended
        Console.Title = "Georgetown Dry Cleaning Services";
        Console.WriteLine("-/- Georgetown Dry Cleaning Services -/-");
        // Request order information from the user
        Console.Write("Enter Customer Name: ");
        customerName = Console.ReadLine();
```

```

Console.Write("Enter Customer Phone: ");
homePhone = Console.ReadLine();
Console.WriteLine("Enter the order date and " +
    "time (mm/dd/yyyyhh:mm AM/PM)");
orderDate = DateTime.Parse(Console.ReadLine());
    // Request the quantity of each category of items
Console.Write("Number of Shirts: ");
numberOfShirts = uint.Parse(Console.ReadLine());
Console.Write("Number of Pants: ");
numberOfPants = uint.Parse(Console.ReadLine());
Console.Write("Number of Dresses: ");
numberOfDresses = uint.Parse(Console.ReadLine());
    // Perform the necessary calculations
subTotalShirts = numberOfShirts * PriceOneShirt;
subTotalPants = numberOfPants * PriceAPairOfPants;
subTotalDresses = numberOfDresses * PriceOneDress;
    // Calculate the "temporary" total of the order
totalOrder = subTotalShirts +
subTotalPants + subTotalDresses;
    // Calculate the tax amount using a constant rate
taxAmount = totalOrder * TaxRate;
    // Add the tax amount to the total order
salesTotal = totalOrder + taxAmount;
    // Communicate the total to the user...
Console.Write("\nThe Total order is: ");
Console.WriteLine(salesTotal);
    // and request money for the order
Console.Write("Amount Tended? ");
amountTended = double.Parse(Console.ReadLine());

```

```

        // Calculate the difference owed to the customer
        // or that the customer still owes to the store
        Difference = amountTended - salesTotal;

    Console.Clear();
    // Display the receipt
    Console.WriteLine("=====");
    Console.WriteLine("-/- Georgetown Dry Cleaning Services -/-");
    Console.WriteLine("=====");
    customerName); Console.WriteLine("Customer: {0}"
    homePhone); Console.WriteLine("Home Phone: {0}"
    orderDate); Console.WriteLine("Date & Time: {0}"
        Console.WriteLine("-----");
        Console.WriteLine("Item Type Qty Unit/Price Sub-Total");
        Console.WriteLine("-----");
        Console.WriteLine("Shirts {0} {1:C} {2}"
        PriceOneShirt * numberOfShirts.ToString()
        subTotalShirts.ToString("C"));
        Console.WriteLine("Pants {0} {1:C} {2:C}"
        PriceAPairOfPants * numberOfPants
        subTotalPants);
        Console.WriteLine("Dresses {0} {1:C} {2:C}"
        PriceOneDress * numberOfDresses
        subTotalDresses);
        Console.WriteLine("-----");
    totalOrder); Console.WriteLine("Total Order: {0:C}"
    TaxRate); Console.WriteLine("Tax Rate: {0:P}"
        Console.WriteLine("Tax Amount: {0}"
        taxAmount.ToString("C"));

```

```

salesTotal); Console.WriteLine("Net Price: {0:F}"
    Console.WriteLine("-----");
amountTended); Console.WriteLine("Amount Tended: {0:C}"
Difference); Console.WriteLine("Difference: {0:C}"
    Console.WriteLine("=====");
    System.Console.ReadKey();
    return 0;
    }
}

```

2. برنامه را اجرا کنید.

```

-/- Georgetown Dry Cleaning Services -/-
Enter Customer Name: Gretchen McCormack
Enter Customer Phone: (410) 739-2884
Enter the order date and time (mm/dd/yyyyhh:mm AM/PM)
04/09/2001 10:25 AM
Number of Shirts: 5
Number of Pants: 12
Number of Dresses: 8

The Total order is: 80.951625
Amount Tended? 100

```

3. حال Enter را بزنید.

```

=====
-/- Georgetown Dry Cleaning Services -/-
=====
Customer: Gretchen McCormack
Home Phone: (410) 739-2884
Date & Time: 4/9/2001 10:25:00 AM
-----
Item Type Qty Unit/Price Sub-Total
-----
Shirts 5 $0.95 $4.75
Pants 12 $2.95 $35.40
Dresses 8 $4.55 $36.40
-----
Total Order: $76.55

```

```

Tax Rate:          5.75 %
Tax Amount:       $4.40
Net Price:        80.95
-----
Amount Tended:   $100.00
Difference:      $19.05
=====

```

4. پنجره ی DOS را ببندید.

قالب بندی خط

در برنامه های فوق به منظور نمایش خطی از یک متن، از توابع **Write()** یا **WriteLine()** استفاده می کردیم. برای قرار دادن متنی با طول متفاوت بالای متن دیگری، با استفاده ی بیش از حد از فضاها ی خالی، یک رشته را خراب می کردیم. چنین تکنیکی تاحدی غیر حرفه ای و نامطمئن تلقی می شود. خوشبختانه، قالب بندی خط و متن، همچنین نحوه ی نمایش آن ها را می توان تاحدی زیادی مدیریت کرد. همان طور که انتظار می رفت، **.NET Framework** به منظور تنظیم مقدار فضای مورد نیاز برای نمایش رشته ای از متن و نحوه ی قرار گیری رشته ی مزبور در خط مربوط به آن مکانیسم های ویژه ای در اختیار کاربر قرار می دهد.

به منظور تعیین مقدار فضای مورد نیاز برای نمایش رشته، می توان از **placeholder** داخل پرانتز **Write()** یا **WriteLine()** استفاده کرد. برای این منظور، داخل **placeholder**، **0** یا تعدادی عدد تصاعدی (**incrementing number**) را به همراه کاراکتر قالب بندی آن (**formatting character**) تایپ کنید. سپس، ویرگول و به دنبال آن تعداد کاراکترهایی که با فضای (پهنا) دلخواه همخوانی دارد / برابر است را وارد کنید.

مثال

```

using System;
public class Exercise
{
    public static void Main()
    {
        var fullName = "Anselme Bogos";
        var age = 15;
        var hSalary = 22.74;
        Console.WriteLine("Full Name: {0,20}",
        fullName);
    }
}

```

```
        Console.WriteLine("Age: {0,14}", age.ToString());
        Console.WriteLine("Distance: {0:C,8}",
hSalary.ToString());
        Console.WriteLine();
    }
}
```

نتیجه

```
Full Name:      Anselme Bogos
Age:           15
Distance:     22.74
```

علامتی که برای پهنا (**width**) در نظر می‌گیرید بسیار مهم است. اگر مثبت بود، خط متن در سمت راست قرار می‌گیرد که باید **alignment** مرجع، به خصوص برای مقادیر عددی باشد. حال چنانچه علامتی که ارائه می‌دهید منفی باشد، متن در سمت چپ قرار می‌گیرد.

قالب بندی تاریخ و زمان

همان طور که پیش تر ذکر شد، هنگامی که کاربر مقدار زمان برای متغیر **DateTime** وارد می‌کند، **compiler** خود یک بخش زمان به مقدار مذکور اضافه می‌کند. اگر بخواهید تنها بخش زمان یا بخش تاریخ بررسی شود، لازم است این امر را برای **compiler** مشخص کنید. برای این منظور، نوع داده‌ی **DateTime** یک سری حروف در نظر گرفته است که با استفاده از آن می‌توان نحوه‌ی نمایش مقدار (نوع قالب بندی آن) برای کاربر را تعیین کرد. کاراکتر مورد نظر داخل **placeholder** متغیر **DateTime** پس از 0 یا مقدار عددی افزایشی (**incremental numeric value**) باید تایپ شود.