

**UNIVERSIDAD AMERICANA**  
**Facultad de Ingeniería y Arquitectura**

---



---

**Metodología y Programación Estructurada**  
**Guia No.2**

---

**Estudiantes**

Kristel Geraldine Villalta Porras. Cif: 23020027

**Fecha:**

Lunes 16 de Septiembre del 2024

**Managua, Nicaragua**

Program.cs

ConsoleApp1

Program

Mail

```
1  /*
2  Ejercicio 1: Simulación de tirar dos dados
3
4  Escriba un programa en C#, que simule el tirar dos dados. El programa deberá utilizar la función
5  rand para tirar el primer dado, y después volverá a utilizar la función rand para tirar el segundo.
6  La suma de los valores deberá entonces ser calculada. Nota: en vista de que cada dado puede
7  mostrar un valor entero de 1 a 6, entonces la suma de los dos valores variará desde 2 hasta 12,
8  siendo 7 la suma más frecuente y 2 y 12 las menos frecuentes. Su programa deberá tirar 36,000
9  veces los dos dados. Utilice un arreglo de un subíndice para llevar cuenta del número de veces que aparece
10 cada suma posible. Imprima los resultados en un formato tabular. También determine si los totales
11 son razonables, por ejemplo, existen seis formas de llegar a un 7, por lo que aproximadamente una
12 sexta parte de todas las tiradas deberán ser 7.
13 */
14
15 using System;
16
17 public class Program
18 {
19     // 0 referencias
20     public static void Main(string[] args)
21     {
22         // 0 referencias
23         Random rand = new Random();
24         int[] sumFrequencies = new int[13];
25
26         // Tirar los dados 36,000 veces
27         for (int i = 0; i < 36000; i++)
28         {
29             int dado1 = rand.Next(1, 7);
30             int dado2 = rand.Next(1, 7);
31             int suma = dado1 + dado2;
32
33             sumFrequencies[suma]++;
34         }
35
36         Console.WriteLine("Suma\tFrecuencia");
37         for (int i = 2; i < sumFrequencies.Length; i++)
38         {
39             Console.WriteLine($"{i}\t{sumFrequencies[i]}");
40         }
41     }
42 }
```



Consola de depuración d ×



Suma	Frecuencia
------	------------

2	1005
---	------

3	2035
---	------

4	2931
---	------

5	3938
---	------

6	5015
---	------

7	6090
---	------

8	5039
---	------

9	4026
---	------

10	2957
----	------

11	1990
----	------

12	974
----	-----

C:\Users\geral\source\repos\ConsoleAp

igo 0.

Presione cualquier tecla para cerrar

```
Program.cs*  Program
ConsoleApp1
1
2  /*
3  Ejercicio 2: Sistema de Reservas de Aerolínea
4
5  Una pequeña aerolínea acaba de adquirir una computadora para su sistema automatizado de reservas.
6  Escriba un programa que asigne asientos en un vuelo con capacidad para 10 asientos. El programa
7  deberá mostrar un menú de alternativas:
8
9  1. Si el usuario digita 1, se asigna un asiento en la sección de fumar (asientos 1 al 5).
10  2. Si el usuario digita 2, se asigna un asiento en la sección de no fumar (asientos 6 al 10).
11
12  El programa imprimirá un pase de abordaje indicando el número de asiento y la sección.
13  Si la sección elegida está llena, deberá preguntar si desea ser asignado en la otra sección.
14  Si se rehúsa, se imprimirá el mensaje "Next flight leaves in 3 hours" y el programa terminará.
15  */
16
17  using System;
18
19  0 referencias
20  public class Program
21  {
22      0 referencias
23      public static void Main(string[] args)
24      {
25          bool[] asientos = new bool[10]; // Arreglo que indica si un asiento está ocupado
26          bool asignado;
27
28          while (true)
29          {
30              Console.WriteLine("Por favor, elija su opción:");
31              Console.WriteLine("1 - Sección de fumar (asientos 1-5)");
32              Console.WriteLine("2 - Sección de no fumar (asientos 6-10)");
33              int opcion = int.Parse(Console.ReadLine());
34
35              asignado = false; // Reiniciamos la variable de asignación
36
37              if (opcion == 1) // Opción de fumar
38              {
39                  // Intentar asignar asiento en la sección de fumar (1 a 5)
40                  for (int i = 0; i < 5; i++)
41                  {
42                      if (!asientos[i])
43                      {
44                          asientos[i] = true;
45                          Console.WriteLine($"Asiento asignado en la sección de fumar: {i + 1}");
46                          asignado = true;
47                          break;
48                      }
49                  }
50
51                  // Si no hay asientos en la sección de fumar
52                  if (!asignado)
53                  {
54                      Console.WriteLine("Sección de fumar llena. ¿Le gustaría un asiento en la sección de no fumar? (S/N)");
55                      string respuesta = Console.ReadLine();
56
57                      if (respuesta.ToUpper() == "S")
58                      {
59                          for (int i = 5; i < 10; i++)
60                          {
61                              if (!asientos[i])
62                              {
63                                  asientos[i] = true;
64                                  Console.WriteLine($"Asiento asignado en la sección de no fumar: {i + 1}");
65                                  asignado = true;
66                              }
67                          }
68                      }
69                  }
70              }
71          }
72      }
73  }
```

```

53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111

```

```

    if (respuesta.ToUpper() == "S")
    {
        for (int i = 5; i < 10; i++)
        {
            if (!asientos[i])
            {
                asientos[i] = true;
                Console.WriteLine($"Asiento asignado en la sección de no fumar: {i + 1}");
                asignado = true;
                break;
            }
        }
    }
    else if (respuesta.ToUpper() == "N")
    {
        Console.WriteLine("Next flight leaves in 3 hours.");
        break; // Termina el programa si la respuesta es "N"
    }
}

else if (opcion == 2) // Opción de no fumar
{
    // Intentar asignar asiento en la sección de no fumar (6 a 10)
    for (int i = 5; i < 10; i++)
    {
        if (!asientos[i])
        {
            asientos[i] = true;
            Console.WriteLine($"Asiento asignado en la sección de no fumar: {i + 1}");
            asignado = true;
            break;
        }
    }

    // Si no hay asientos en la sección de no fumar
    if (!asignado)
    {
        Console.WriteLine("Sección de no fumar llena. ¿Le gustaría un asiento en la sección de fumar? (S/N)");
        string respuesta = Console.ReadLine();

        if (respuesta.ToUpper() == "S")
        {
            for (int i = 0; i < 5; i++)
            {
                if (!asientos[i])
                {
                    asientos[i] = true;
                    Console.WriteLine($"Asiento asignado en la sección de fumar: {i + 1}");
                    asignado = true;
                    break;
                }
            }
        }
        else if (respuesta.ToUpper() == "N")
        {
            Console.WriteLine("Next flight leaves in 3 hours.");
            break; // Termina el programa si la respuesta es "N"
        }
    }
}

```

```
Program.cs  X
ConsoleApp1  Program
74  else if (opcion == 2) // Opción de no fumar
75  {
76      // Intentar asignar asiento en la sección de no fumar (6 a 10)
77      for (int i = 5; i < 10; i++)
78      {
79          if (!asientos[i])
80          {
81              asientos[i] = true;
82              Console.WriteLine($"Asiento asignado en la sección de no fumar: {i + 1}");
83              asignado = true;
84              break;
85          }
86      }
87
88      // Si no hay asientos en la sección de no fumar
89      if (!asignado)
90      {
91          Console.WriteLine("Sección de no fumar llena. ¿Le gustaría un asiento en la sección de fumar? (S/N)");
92          string respuesta = Console.ReadLine();
93
94          if (respuesta.ToUpper() == "S")
95          {
96              for (int i = 0; i < 5; i++)
97              {
98                  if (!asientos[i])
99                  {
100                      asientos[i] = true;
101                      Console.WriteLine($"Asiento asignado en la sección de fumar: {i + 1}");
102                      asignado = true;
103                      break;
104                  }
105              }
106          }
107          else if (respuesta.ToUpper() == "N")
108          {
109              Console.WriteLine("Next flight leaves in 3 hours.");
110              break; // Termina el programa si la respuesta es "N"
111          }
112      }
113
114
115      // Si no se pudo asignar ningún asiento
116      if (!asignado)
117      {
118          Console.WriteLine("Next flight leaves in 3 hours.");
119          break; // Finaliza el programa si no hay asignación posible
120      }
121  }
122  }
123  }
124  }
```

```
Consola de depuración d  X  +  v
1 - Sección de fumar (asientos 1-5)
2 - Sección de no fumar (asientos 6-10)
1
Asiento asignado en la sección de fumar: 2
Por favor, elija su opción:
1 - Sección de fumar (asientos 1-5)
2 - Sección de no fumar (asientos 6-10)
1
Asiento asignado en la sección de fumar: 3
Por favor, elija su opción:
1 - Sección de fumar (asientos 1-5)
2 - Sección de no fumar (asientos 6-10)
1
Asiento asignado en la sección de fumar: 4
Por favor, elija su opción:
1 - Sección de fumar (asientos 1-5)
2 - Sección de no fumar (asientos 6-10)
1
Asiento asignado en la sección de fumar: 5
Por favor, elija su opción:
1 - Sección de fumar (asientos 1-5)
2 - Sección de no fumar (asientos 6-10)
1
Sección de fumar llena. ¿Le gustaría un asiento en la sección de no fumar? (S/N)
N
Next flight leaves in 3 hours.

C:\Users\geral\source\repos\ConsoleApp1\ConsoleApp1\bin\Debug\net8.0\ConsoleApp1.exe (proceso 23980) se cerró con el código 0.
Presione cualquier tecla para cerrar esta ventana. . . |
```

Program.cs\* [icon] [X]

C# ConsoleApp1

Program

```
1  /*
2  Ejercicio 3: Resumen de Ventas por Vendedor y Producto
3
4  Una empresa tiene 4 vendedores (1 a 4) que venden 5 productos diferentes (1 a 5).
5  Cada vendedor emite un reporte diario por cada tipo de producto vendido, que incluye el número
6  del vendedor, el número del producto y el valor total en dólares vendido ese día.
7
8  Escriba un programa que lea la información de ventas del mes anterior y resuma las ventas totales
9  por vendedor y por producto. Los resultados deberán almacenarse en un arreglo bidimensional y
10 se imprimirán en formato tabular.
11 */
12
13 using System;
14
15 0 referencias
16 public class Program
17 {
18     0 referencias
19     public static void Main(string[] args)
20     {
21         decimal[,] ventas = new decimal[5, 6];
22
23         Console.WriteLine("Ingrese la información de ventas: ");
24         for (int i = 0; i < 5; i++)
25         { // aqui asigno que max sean 5 productos
26             for (int j = 0; j < 4; j++) // aqui asigno que max son 4 vendedores
27             {
28                 Console.Write($"Ventas del producto {i + 1} por el vendedor {j + 1}: ");
29                 ventas[i, j] = decimal.Parse(Console.ReadLine());
30                 ventas[i, 4] += ventas[i, j]; // Sumar ventas por producto
31                 ventas[4, j] += ventas[i, j]; // Sumar ventas por vendedor
32             }
33         }
34         Console.WriteLine("\nResumen de ventas :");
35         Console.WriteLine("Producto\tVendedor 1\tVendedor 2\tVendedor 3\tVendedor 4\tTotal Producto");
36         for (int i = 0; i < 5; i++)
37         {
38             Console.Write($"Producto {i + 1}\t");
39             for (int j = 0; j < 5; j++)
40             {
41                 Console.Write($"{ventas[i, j]}\t\t");
42             }
43             Console.WriteLine();
44         }
45         Console.WriteLine("Total Vendedor\t");
46         for (int j = 0; j < 4; j++)
47         {
48             Console.Write($"{ventas[4, j]}\t\t");
49         }
50         Console.WriteLine();
51     }
52 }
```

Consola de depuración d +

Ventas del producto 2 por el vendedor 1: 3  
Ventas del producto 2 por el vendedor 2: 1  
Ventas del producto 2 por el vendedor 3: 2  
Ventas del producto 2 por el vendedor 4: 4  
Ventas del producto 3 por el vendedor 1: 3  
Ventas del producto 3 por el vendedor 2: 2  
Ventas del producto 3 por el vendedor 3: 5  
Ventas del producto 3 por el vendedor 4: 2  
Ventas del producto 4 por el vendedor 1: 2  
Ventas del producto 4 por el vendedor 2: 3  
Ventas del producto 4 por el vendedor 3: 4  
Ventas del producto 4 por el vendedor 4: 5  
Ventas del producto 5 por el vendedor 1: 2  
Ventas del producto 5 por el vendedor 2: 2  
Ventas del producto 5 por el vendedor 3: 4  
Ventas del producto 5 por el vendedor 4: 5

Resumen de ventas (en dólares):

Producto	Vendedor 1	Vendedor 2	Vendedor 3	Vendedor 4	Total Producto
Producto 1	3	2	5	4	14
Producto 2	3	1	2	4	10
Producto 3	3	2	5	2	12
Producto 4	2	3	4	5	14
Producto 5	4	4	8	10	13
Total Vendedor	4	8	10		

C:\Users\geral\source\repos\ConsoleApp1\ConsoleApp1\bin\Debug\net8.0\ConsoleApp1.exe (proceso 4788) se cerró con el código 0.  
Presione cualquier tecla para cerrar esta ventana. . .|



Program.cs\*

ConsoleApp1

ProgramaVentas

```
1  //Ejercicio 4
2  // Instrucciones:
3  // 1. Este programa calcula el salario semanal de los vendedores basado en un salario base
4  //    de $200 más una comisión del 9% sobre las ventas brutas.
5  // 2. Luego, clasifica el salario de los vendedores en diferentes rangos y cuenta cuántos
6  //    vendedores caen en cada uno de esos rangos.
7  // 3. El usuario puede ingresar los montos de ventas para cada vendedor.
8  // 4. El salario se trunca a una cantidad entera antes de ser clasificado en los rangos.
9  // 5. Los rangos de salario son:
10 //    - $200-$299
11 //    - $300-$399
12 //    - $400-$499
13 //    - $500-$599
14 //    - $600-$699
15 //    - $700-$799
16 //    - $800-$899
17 //    - $900-$999
18 //    - $1000 o superior
19
20 using System;
21
22 0 referencias
23 public class ProgramaVentas
24 {
25     0 referencias
26     public static void Main(string[] args)
27     {
28         // Arreglo de contadores para cada rango de salarios
29         int[] contadores = new int[9];
30
31         // Solicitar el número de vendedores
32         Console.WriteLine("Ingresa el número de vendedores: ");
33         int numVendedores = int.Parse(Console.ReadLine());
34
35         // Procesar cada vendedor
36         for (int i = 0; i < numVendedores; i++)
37         {
38             // Solicitar las ventas brutas de cada vendedor
39             Console.WriteLine($"Ingresa las ventas brutas del vendedor {i + 1}: ");
40             double ventas = double.Parse(Console.ReadLine());
41
42             // Calcular salario total: $200 + 9% de las ventas brutas
43             double salario = 200 + (ventas * 0.09);
44
45             // Truncar a cantidad entera
46             int salarioEntero = (int)salario;
47
48             // Determinar en qué rango cae el salario
49             if (salarioEntero >= 1000)
```

74 %

0

2

Salida

Lista de errores

Salida

Compilación correcta

Program.cs\*

ConsoleApp1

```
41 double salario = 200 + (ventas * 0.09);
42
43 // Truncar a cantidad entera
44 int salarioEntero = (int)salario;
45
46 // Determinar en qué rango cae el salario
47 if (salarioEntero >= 1000)
48 {
49     contadores[8]++; // Rango $1000 o superior
50 }
51 else if (salarioEntero >= 900)
52 {
53     contadores[7]++; // Rango $900-$999
54 }
55 else if (salarioEntero >= 800)
56 {
57     contadores[6]++; // Rango $800-$899
58 }
59 else if (salarioEntero >= 700)
60 {
61     contadores[5]++; // Rango $700-$799
62 }
63 else if (salarioEntero >= 600)
64 {
65     contadores[4]++; // Rango $600-$699
66 }
67 else if (salarioEntero >= 500)
68 {
69     contadores[3]++; // Rango $500-$599
70 }
71 else if (salarioEntero >= 400)
72 {
73     contadores[2]++; // Rango $400-$499
74 }
75 else if (salarioEntero >= 300)
76 {
77     contadores[1]++; // Rango $300-$399
78 }
79 else
80 {
81     contadores[0]++; // Rango $200-$299
82 }
83
84
85 // Mostrar los resultados
86 Console.WriteLine("\nRangos de salarios:");
87 Console.WriteLine("$200-$299: " + contadores[0]);
88 Console.WriteLine("$300-$399: " + contadores[1]);
89 Console.WriteLine("$400-$499: " + contadores[2]);
90 Console.WriteLine("$500-$599: " + contadores[3]);
91 Console.WriteLine("$600-$699: " + contadores[4]);
92 Console.WriteLine("$700-$799: " + contadores[5]);
93 Console.WriteLine("$800-$899: " + contadores[6]);
94 Console.WriteLine("$900-$999: " + contadores[7]);
95 Console.WriteLine("$1000 o superior: " + contadores[8]);
96
97
98 }
```

61 %



0

2



Consola de depuración d × + ▾

Ingresa el número de vendedores: 5

Ingresa las ventas brutas del vendedor 1: 1235

Ingresa las ventas brutas del vendedor 2: 8888

Ingresa las ventas brutas del vendedor 3: 43221

Ingresa las ventas brutas del vendedor 4: 2567

Ingresa las ventas brutas del vendedor 5: 3321

Rangos de salarios:

\$200-\$299: 0

\$300-\$399: 1

\$400-\$499: 2

\$500-\$599: 0

\$600-\$699: 0

\$700-\$799: 0

\$800-\$899: 0

\$900-\$999: 1

\$1000 o superior: 1

C:\Users\geral\source\repos\ConsoleApp1\ConsoleApp1\bin\Debug\net8.0\ConsoleApp1.exe  
igo 0.

Presione cualquier tecla para cerrar esta ventana. . .|