Capgemini training – Assignment 1(Day 2)

```
using System;

class Program
{
    static void Main()
    {
        Exercise1();
        Exercise2();
        Exercise3();
        Exercise4();
        Exercise5();
        Exercise6();
        Exercise7();
        Exercise8();
        Exercise9();
        Exercise10();
    }

    // Exercise 1: Student Attendance & Eligibility
    static void Exercise1()
    {
        Console.Write("Enter classes attended: ");
        int attended = int.Parse(Console.ReadLine());

        Console.Write("Enter total classes: ");
    int total = int.Parse(Console.ReadLine());

    double percentage = (double)attended / total * 100;
    int displayPercentage = (int)Math.Round(percentage);

    Console.WriteLine($"Attendance Percentage: {displayPercentage}%");
}

// Exercise 2: Online Examination Result Processing
static void Exercise2()
{
    Console.Write("Enter marks 1: ");
    int m1 = int.Parse(Console.ReadLine());

    Console.Write("Enter marks 2: ");
    int m2 = int.Parse(Console.ReadLine());

    Console.Write("Enter marks 3: ");
    int m3 = int.Parse(Console.ReadLine());

    double average = (m1 + m2 + m3) / 3.0;
    Console.WriteLine($"Average: {average:F2}");

    int scholarshipScore = (int)Math.Round(average);
    Console.WriteLine($"Scholarship Score: {scholarshipScore}");
}
```

```
// Exercise 3: Library Fine Calculation
static void Exercise3()
{
    Console.Write("Enter fine per day: ");
    decimal finePerDay = decimal.Parse(Console.ReadLine());

    Console.Write("Enter days overdue: ");
    int daysLate = int.Parse(Console.ReadLine());

    decimal totalFine = finePerDay * daysLate;
    double analyticsFine = (double)totalFine;

    Console.WriteLine($"Total Fine: {totalFine}");
    Console.WriteLine($"Logged Fine (double): {analyticsFine}");
}

// Exercise 4: Banking Interest Calculation
static void Exercise4()
{
    Console.Write("Enter account balance: ");
    decimal balance = decimal.Parse(Console.ReadLine());

    Console.Write("Enter interest rate (%): ");
    float interestRate = float.Parse(Console.ReadLine());

    decimal interest = balance * (decimal)interestRate / 100;
    balance += interest;
    Console.WriteLine($"Updated Balance: {balance}");
}

// Exercise 5: E-Commerce Order Pricing
static void Exercise5()
{
    Console.Write("Enter cart total: ");
    double cartTotal = double.Parse(Console.ReadLine());

    Console.Write("Enter tax rate (decimal): ");
    decimal taxRate = decimal.Parse(Console.ReadLine());

    Console.Write("Enter discount: ");
    decimal discount = decimal.Parse(Console.ReadLine());

    decimal finalAmount =
        (decimal)cartTotal +
        ((decimal)cartTotal * taxRate) -
        discount;

    Console.WriteLine($"Final Payable Amount: {finalAmount}");
}

// Exercise 6: Weather Monitoring
static void Exercise6()
{
    Console.Write("Enter temperature in Kelvin: ");
    short sensorReading = short.Parse(Console.ReadLine());

    double celsius = sensorReading - 273.15;
    int displayTemp = (int)Math.Round(celsius);

    Console.WriteLine($"Temperature in Celsius: {displayTemp}°C");
}

// Exercise 7: University Grading Engine
static void Exercise7()
{
    Console.Write("Enter final score: ");
    double finalScore = double.Parse(Console.ReadLine());

    byte grade;

    if (finalScore >= 90) grade = 10;
    else if (finalScore >= 80) grade = 9;
    else if (finalScore >= 70) grade = 8;
    else grade = 7;

    Console.WriteLine($"Grade Point: {grade}");
}

// Exercise 8: Mobile Data Usage Tracker
static void Exercise8()
```

```
{
   Console.Write("Enter data usage in bytes: ");
   long bytesUsed = long.Parse(Console.ReadLine());

   double mb = bytesUsed / (1024.0 * 1024);
   double gb = bytesUsed / (1024.0 * 1024 * 1024);

   int roundedGB = (int)Math.Round(gb);

   Console.WriteLine($"Usage: {mb:F2} MB");
   Console.WriteLine($"Rounded Usage: {roundedGB} GB");
}

// Exercise 9: Warehouse Inventory Capacity
static void Exercise9()
{
   Console.Write("Enter current item count: ");
   int items = int.Parse(Console.ReadLine());

   Console.Write("Enter maximum capacity: ");
   ushort capacity = ushort.Parse(Console.ReadLine());

   bool isOverLimit = items > capacity;
   Console.WriteLine($"Over Capacity: {isOverLimit}");
}

// Exercise 10: Payroll Salary Computation
```

```
    static void Exercise10()
    {
        Console.Write("Enter basic salary: ");
        int basicSalary = int.Parse(Console.ReadLine());

        Console.Write("Enter allowance: ");
        double allowance = double.Parse(Console.ReadLine());

        Console.Write("Enter deduction: ");
        double deduction = double.Parse(Console.ReadLine());

        decimal netSalary =
            basicSalary +
            (decimal)allowance -
            (decimal)deduction;

        Console.WriteLine($"Net Salary: {netSalary}");
    }
}
```