Day2: Work

```
    Exercise1();

    Exercise2();

    Exercise3();

    Exercise4();

    Exercise5();

    Exercise6();

    Exercise7();

    Exercise8();

    Exercise9();

    Exercise10();
static void Exercise1()

{

int TotalDays=100;

int ClassesAttended=88;

double AttendancePercentage=(ClassesAttended/(double)TotalDays)*100;

Console.WriteLine($"[Ex1] Attendance Percentage: {AttendancePercentage}%");

}
 static void Exercise2()

    {

        int subject1 = 85, subject2 = 90, subject3 = 78;

        double average = (subject1 + subject2 + subject3) / 3.0;

        Console.WriteLine($"[Ex2] Average (2 dp): {average:F2}");


        int scholarshipScore = (int)Math.Round(average);

        Console.WriteLine($"[Ex2] Scholarship Score (int): {scholarshipScore}");

    }
```

```csharp
static void Exercise3()
{
    decimal finePerDay = 2.50m;
    int daysOverdue = 4;

    decimal totalFine = finePerDay * daysOverdue;
    double fineForAnalytics = (double)totalFine;

    Console.WriteLine($"[Ex3] Total Fine (decimal): {totalFine}");
    Console.WriteLine($"[Ex3] Fine for Analytics (double): {fineForAnalytics}");
}

static void Exercise4()
{
    decimal balance = 10000m;
    float interestRate = 5.5f;

    decimal rateAsDecimal = (decimal)interestRate;
    decimal monthlyInterest = balance * rateAsDecimal / 100 / 12;

    balance += monthlyInterest;

    Console.WriteLine($"[Ex4] Updated Balance: {balance}");
}

static void Exercise5()
{
    double cartTotal = 199.99;
    decimal taxRate = 0.18m;
    decimal discount = 20m;
```

```csharp
        decimal totalAsDecimal = (decimal)cartTotal;

        decimal tax = totalAsDecimal * taxRate;


        decimal finalAmount = totalAsDecimal + tax - discount;


        Console.WriteLine($"[Ex5] Final Payable Amount: {finalAmount}");
    }


    static void Exercise6()
    {
        short sensorReading = 300;
        double celsius = sensorReading / 10.0;


        int displayTemp = (int)Math.Round(celsius);


        Console.WriteLine($"[Ex6] Temperature: {displayTemp}°C");
    }


    static void Exercise7()
    {
        double finalScore = 82.6;


        byte grade;


        if (finalScore >= 85) grade = 1;
        else if (finalScore >= 70) grade = 2;
        else if (finalScore >= 50) grade = 3;
        else grade = 4;


        Console.WriteLine($"[Ex7] Grade Code: {grade}");
```

```csharp
    }

    static void Exercise8()
    {
        long bytesUsed = 5_368_709_120;

        double mb = bytesUsed / 1024.0 / 1024.0;
        double gb = bytesUsed / 1024.0 / 1024.0 / 1024.0;

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

        Console.WriteLine($"[Ex8] Usage: {mb:F2} MB | {gb:F2} GB | Rounded: {roundedGb} GB");
    }

    static void Exercise9()
    {
        int itemCount = 500;
        ushort maxCapacity = 450;

        bool isOverCapacity = itemCount > maxCapacity;

        Console.WriteLine($"[Ex9] Over Capacity: {isOverCapacity}");
    }

    static void Exercise10()
    {
        int basicSalary = 30000;
        double allowances = 5500.75;
        double deductions = 2200.50;

        decimal netSalary = basicSalary + (decimal)allowances - (decimal)deductions;
```
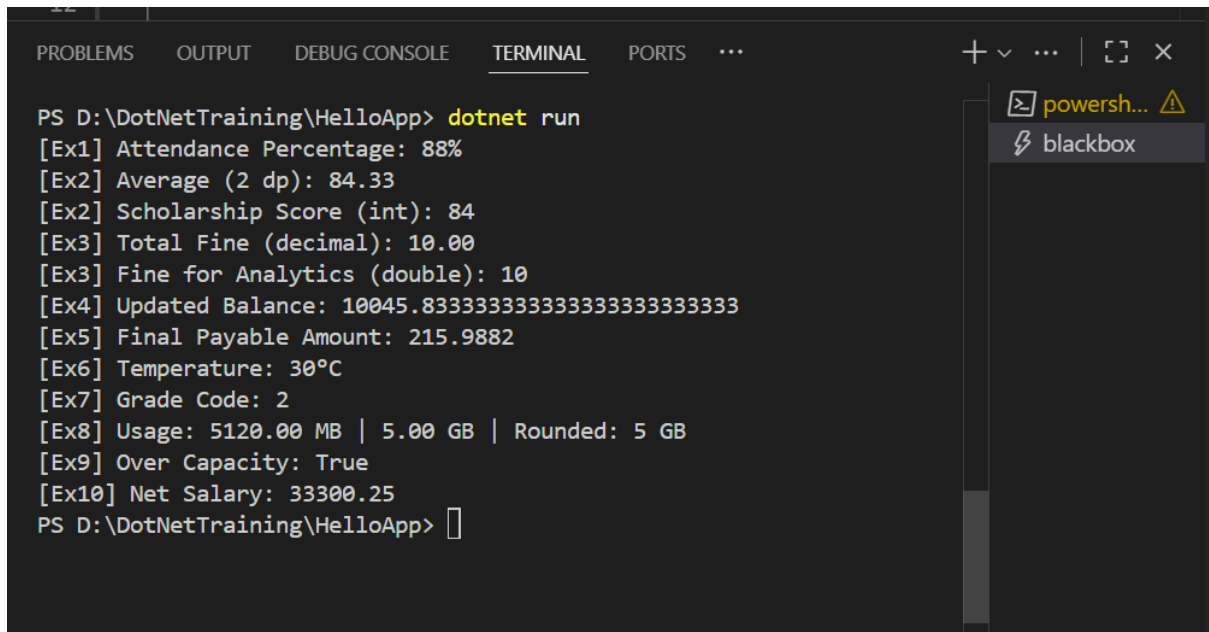
```
        Console.WriteLine($"[Ex10] Net Salary: {netSalary}");
    }
```

Output:

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS    ...

PS D:\DotNetTraining\HelloApp> dotnet run
[Ex1] Attendance Percentage: 88%
[Ex2] Average (2 dp): 84.33
[Ex2] Scholarship Score (int): 84
[Ex3] Total Fine (decimal): 10.00
[Ex3] Fine for Analytics (double): 10
[Ex4] Updated Balance: 10045.8333333333333333333333
[Ex5] Final Payable Amount: 215.9882
[Ex6] Temperature: 30°C
[Ex7] Grade Code: 2
[Ex8] Usage: 5120.00 MB | 5.00 GB | Rounded: 5 GB
[Ex9] Over Capacity: True
[Ex10] Net Salary: 33300.25
PS D:\DotNetTraining\HelloApp>
```