# Exercises: Simple Commands

Test your solutions in the **judge system**: [https://judge.softuni.org/Contests/4624](https://judge.softuni.org/Contests/4624/Simple-Commands-and-Visual-Studio-Exercise)

## Text Reading

Write a **console program** that:

* Reads **input** from the console
* Print the **entered** text on the console

### Example Input / Output

|  |  |
| --- | --- |
| **Input** | **Output** |
| C# Rocks | C# Rocks |

string text = Console.ReadLine();

Console.WriteLine(text);

## Square Area

Write a **console program** that:

* Reads the integer number, which represents **length of one side of a square**
* Calculates **its area**
* Prints the **calculated area** on the console

**Note**: Square's area is calculated when you multiplied length by length: **length \* length**

### Example Input / Output

//

int side = int.Parse(Console.ReadLine());

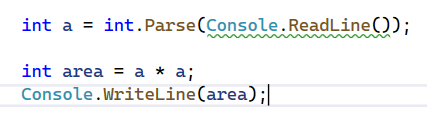
Console.WriteLine(side \* side);

|  |  |
| --- | --- |
| **Input** | **Output** |
| 2 | 4 |

|  |  |
| --- | --- |
| **Input** | **Output** |
| 6 | 36 |

### Instructions

1. **Initialize** an int variable (a) and **assign a value** from the **input from the console**:  
    
2. Initialize a second variable named **area,** in which you will **store the value for the square’s area**, obtained using **the formula a \* a. Print the resulting value.**



//

int side = int.Parse(Console.ReadLine());

int area = side \* side;

Console.WriteLine(area);

## Rectangle Area

Write a **console program** that:

* Reads two integer numbers, which represents **length and width of the rectangle**
* Calculates **rectangle's area**
* Prints the **calculated area** on the console

**Note**: Rectangle's area is calculated when you multiplied length by width: **length \* width**

### Example Input / Output

//

int a = int.Parse(Console.ReadLine());

int b = int.Parse(Console.ReadLine());

Console.WriteLine(a\*b);

|  |  |
| --- | --- |
| **Input** | **Output** |
| 2  5 | 10 |

## Trapezoid Area

Write a **console program** that:

* Reads three integer numbers, which represents **first** **base, second base and height of the trapezoid**
* Calculates **trapezoid's area**
* Prints the **calculated area** on the console

**Note**: Trapezoid's area is calculated when you sum two bases, divide them by two and the result is multiplied by height: **(first base + second base) / 2 \* height**

### Example Input / Output

//

int a = int.Parse(Console.ReadLine());

int b = int.Parse(Console.ReadLine());

int h = int.Parse(Console.ReadLine());

Console.WriteLine(((a+b)/2)\*h);

|  |  |
| --- | --- |
| **Input** | **Output** |
| 6  2  3 | 12 |
| //Input  int firstBase = int.Parse(Console.ReadLine());  int secondBase = int.Parse(Console.ReadLine());  int height = int.Parse(Console.ReadLine());  //Act (Process)  int area = (firstBase + secondBase) / 2 \* height;  //Output  Console.WriteLine(area); |  |

## Triangle Perimeter

Write a **console program** that:

* Reads three integer numbers, which represents **sides of the triangle**
* Calculates **triangle's perimeter**
* Prints the **calculated perimeter** on the console

**Note**: Triangle's perimeter is calculated when you sum all sides values.

### Example Input / Output

|  |  |
| --- | --- |
| **Input** | **Output** |
| 6  2  3 | 11 |

//Input

int sideA = int.Parse(Console.ReadLine());

int sideB = int.Parse(Console.ReadLine());

int sideC = int.Parse(Console.ReadLine());

//Act (Processing)

int perimeter = sideA + sideB + sideC;

//Output

Console.WriteLine(perimeter);

## Inches to Centimeters Converter

Write a console program:

* Reads **a length in inches** from the console
* Converts it to **centimeters**
* Print the **converted length in centimeters on the console**

**Note: For calculation, multiply the inches by 2.54** (1 inch = 2.54 centimeters).

### Example Input/Output

|  |  |
| --- | --- |
| **Input** | **Output** |
| 5 | 12.7 |

|  |  |
| --- | --- |
| **Input** | **Output** |
| 7 | 17.78 |

//Input

double inch = double.Parse(Console.ReadLine());

//Act (Processing)

double centimeters = inch \* 2.54;

//Output

Console.WriteLine(centimeters);

**Attention**: depending on the **regional settings** of the operating system, instead of a **decimal point** (US settings), a **decimal comma** (BG settings) may be used by default. If the program expects a decimal point and a number with a decimal comma is entered, or vice versa (a decimal point is entered when a decimal comma is expected), the following error will occur:  


It is recommended to **adjust your computer settings to use a decimal point**:****

