

Упражнения: Връщане на стойност и предефиниране на методи

1. Конвертор за температури

Създайте метод, който конвертира температура от **Фаренхайт** в **Целзий**. Форматирайте резултата до втория десетичен знак.

Използвайте формулата: $(\text{fahrenheit} - 32) * 5 / 9$.

Примери

Вход	Изход
95	35.00
33.8	1.00
-40	-40.00

Подсказки

- Прочетете входните данни
- Създайте метод, който **връща стойност от тип double**:
- Извикайте метода в main и **запишете върнатата стойност в нова променлива**:

```
static double FahrenheitToCelsius(double fahrenheit)
{
    return (fahrenheit - 32) * 5 / 9;
}
```

```
static void Main(string[] args)
{
    var fahrenheit = double.Parse(Console.ReadLine());
    var celsius = FahrenheitToCelsius(fahrenheit);
    Console.WriteLine("{0:F2}", celsius);
}
```

2. Пресмятане на лице на триъгълник

Създайте метод, който изчислява и **връща** лицето на триъгълник по дадени основа и височина:

Пример

Вход	Изход
3 4	6

Подсказки

- Първо прочетете входните данни
- После създайте метод, но този път **вместо** да пишете **"static void"** преди името му, напишете **"static double"**, така че да го накараме да **върне стойност от тип double**:

```
static double GetTriangleArea(double width, double height)
{
    return width * height / 2;
}
```

3. Извикайте метода в main и съхранете върнатата стойност в нова променлива:

```
static void Main()
{
    double width = double.Parse(Console.ReadLine());
    double height = double.Parse(Console.ReadLine());
    double area = GetTriangleArea(width, height);
    Console.WriteLine(area);
}
```

3. Повдигане на степен

Създайте метод, който пресмята и връща стойността на число, повдигнато на указаната степен:

Примери

Вход	Изход
2 8	256
3 4	81

Подсказки

1. Както обикновено, прочетете входните данни
2. Създайте метод, който ще има два параметъра - числото и степента, и ще връща резултат от тип double:

```
static double RaiseToPower(double number, int power)
{
    double result = 0d;

    // TODO: Calculate result (use a loop, or Math.Pow())

    return result;
}
```

3. Отпечатайте резултата

4. По-голямата от две стойности

Имате подадени като входни данни две стойности от един и същи тип. Стойностите може да са от тип int, char или string. Създайте метод GetMax() който връща по-голямата от двете стойности:

Примери

Вход	Изход
int 2	16

16	
char a z	z
string Ivan Todor	Todor

Подсказки

1. За тази задача ще трябва да създадете три метода с едно и също име и с различни сигнатури
2. Създайте метод, който ще сравнява цели числа:

```
static int GetMax(int first, int second)
{
    if (first >= second)
    {
        // TODO: return value
    }

    // TODO: handle other cases
}
```

3. Създайте втори метод със същото име, който ще сравнява символи. Следвайте логиката на предния метод:

```
static char GetMax(char first, char second)
{
    // TODO: create logic
}
```

4. И накрая създайте метод за сравняване на низове. Той ще е малко по-различен, тъй като низовете не може да бъдат сравнявани с операторите > и <

```
static string GetMax(string first, string second)
{
    if (first.CompareTo(second) >= 0)
    {
        // TODO: return value
    }
    // TODO: return value
}
```

Трябва да използвате метода "CompareTo()", който връща целочислена стойност (положителна ако сравняваният обект е по-голям, отрицателна, ако е по-малък и нула, ако двата са равни).

5. Последната стъпка е да прочетете входните данни, да използвате променливи от подходящ тип и да извикате GetMax() от вашия Main():

```
var type = Console.ReadLine();
if (type == "int")
{
    int first = int.Parse(Console.ReadLine());
    int second = int.Parse(Console.ReadLine());
    int max = GetMax(first, second);
    Console.WriteLine(max);
}
else if (type == "char")
{
    // TODO: call GetMax with char arguments
}
else if (type == "string")
{
    // TODO: call GetMax with string arguments
}
```