

# Лаб: Първи стъпки в програмирането

Задачи за упражнение в клас и за домашно към курса ["Основи на програмирането" @ СофтУни](#).

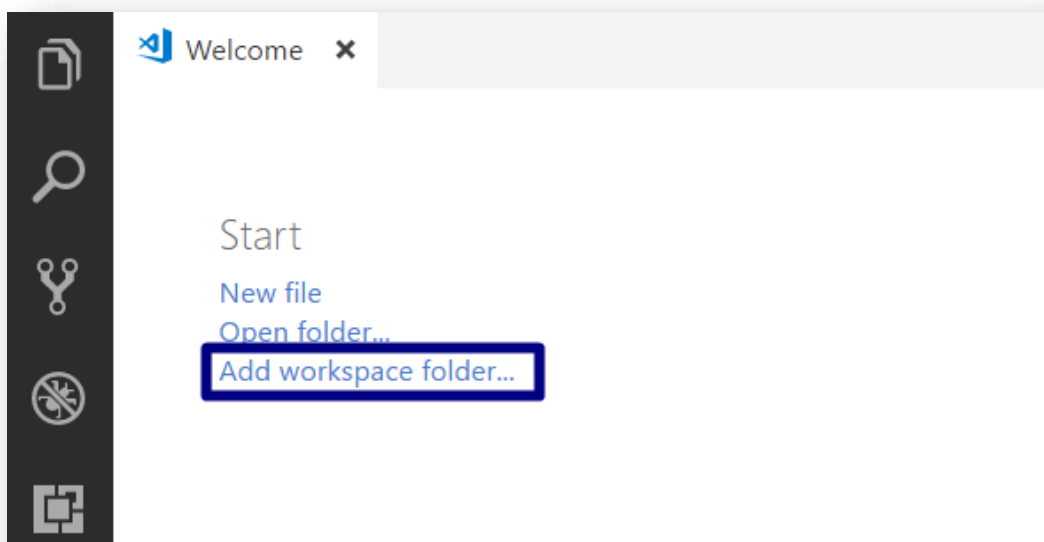
Тествайте решенията си в онлайн **judge** системата:

<https://alpha.judge.softuni.org/contests/first-steps-in-coding-lab/2399>

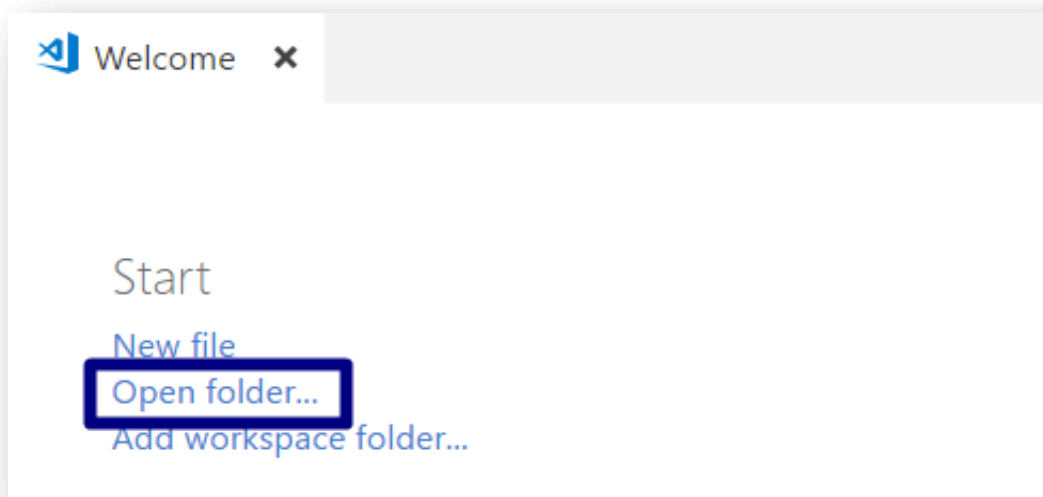
## 0. Празен Visual Studio Code проект

Създайте празен проект във **Visual Studio Code**. Ще обединяваме решенията на всички задачи, под формата на отделни файлове в този проект. Тази възможност е изключително удобна, когато искаме да работим по няколко проекта и бързо да превключваме между тях или искаме да обединим логически няколко взаимосвързани проекта. Това ни помага да запазваме решенията на задачите отделно и да ги пазим, за да ги използваме за други задачи или преговор.

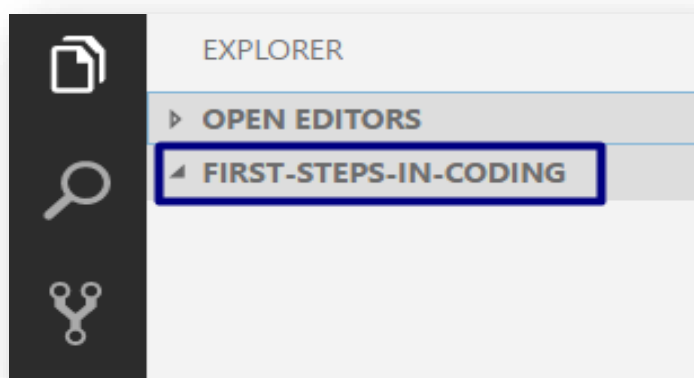
1. Стартирайте Visual Studio Code
2. Създайте нова папка, която ще държи отделните решения. Ще се отвори диалогов прозорец, в който ще трябва да изберете нейната директория. Препоръчително е да именувате папката според темата на заданието, пример **"First-Steps-in-Calculations"**



След това изберете папката като работна среда, за да добавяте файловете с JavaScript решенията на своите задачи в нея.



Панелът в ляво ще изглежда така:

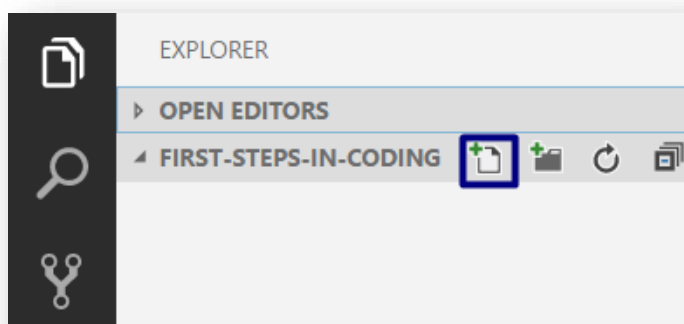


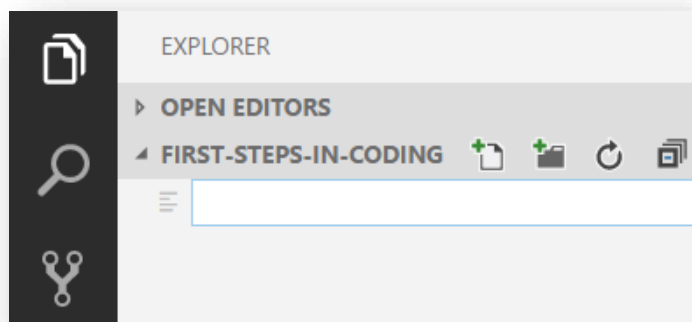
## 1. Конзолна програма "Hello SoftUni"

Напишете **функция**, която отпечатва текста "Hello SoftUni".

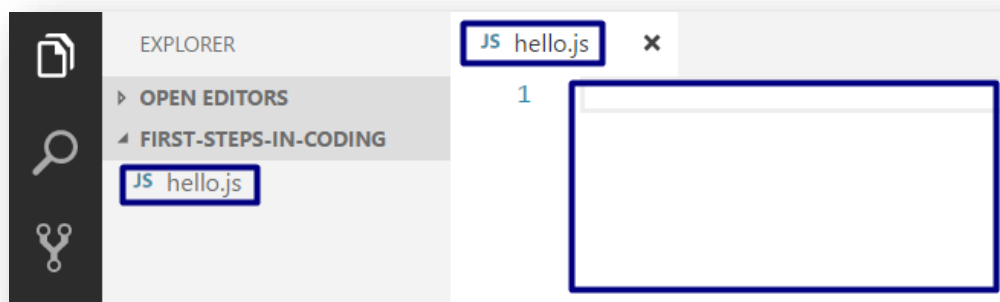
### Насоки

1. Създайте **нов JavaScript файл** в съществуващата папка и го именувайте подходящо. Препоръчително е всеки скриптов файл да се казва както името на задачата, чието решение съдържа.





2. Съдържанието на новият файл ще се отвори в прозореца в **дясно**.



3. Отидете във файла **hello.js** и създайте функцията **hello()**. Можете да си помогнете с картинката по-долу:

```
JS hello.js  x
1  function hello() {
2      console.log("Hello SoftUni")
3  }
```

4. Стартирайте програмата с: **Ctrl + F5**. За да получим резултат трябва да "извикаме" функцията:

```
JS hello.js  x
1  function hello() {
2      console.log("Hello SoftUni")
3  }
4
5  hello()
```

5. Можем да видим резултата в конзолата по-долу:

```
JS hello.js x
1 function hello() {
2   console.log("Hello SoftUni")
3 }
4
5 hello()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

C:\Program Files\nodejs\node.exe hello.js  
Hello SoftUni

6. Тествайте решението на тази задача в онлайн [Judge](#) системата:

## 01. Hello SoftUni

PROBLEM TESTS VIEW DOCS

УСЛОВИЯ

```
1 function hello() {
2   console.log("Hello SoftUni");
3 }
```

Submissions					
ID	From	Time and Memory Used		Result	Strategy
#37657	a few seconds ago	0.69 MB	0.12 s.	✓ 100 / 100	JavaScript code (NodeJS) <a href="#">DETAILS</a>

## 2. Числата от 1 до 10

Напишете функция, която **отпечатва числата от 1 до 10** на отделни редове на конзолата.

### Насоки

1. Създайте **нов JavaScript файл** в съществуващата папка и го именувайте **"Nums-1-To-10"**. Използвайте тялото на функцията **"nums1To10()"**:

```
function nums1To10() {
    // Your code goes here
}

// Call the function to execute the code inside
nums1To10();
```

2. Напишете 10 команди **console.log()**, една след друга, за да отпечатате числата от 1 до 10.

```
function nums1to10() {
    console.log(1);
    console.log(2);
    console.log(3);
    console.log(4);
    console.log(5);
    console.log(6);
    console.log(7);
    console.log(8);
    console.log(9);
    console.log(10);
}
```

3. Стартирайте програмата с: **Ctrl + F5**. За да получите резултат трябва да "извикате" функцията.

### 3. Пресмятане на лице на квадрат

Да се напише **функция**, която **получава цяло число** и **пресмята лицето на квадрат дадената страна**.

#### Примерен вход и изход

вход	изход
5	25

#### Насоки

1. Приемете вход (число).
2. Инициализирайте променлива, която държи пресметнатата стойност на лицето, получена по формулата  $a * a$  и принтирайте резултата, като не забравяте, че за да получите локално резултат, трябва да извикате функцията и да ѝ подадете входни данни.

```
function squareArea(a) {
  let area = a * a;
  console.log(area);
}
```

## 4. Конвертор: инчове към сантиметри

Да се напише функция, която получава **реално число** и го преобразува **от инчове в сантиметри**. За целта **умножете инчовете по 2.54** (1 инч = 2.54 сантиметра).

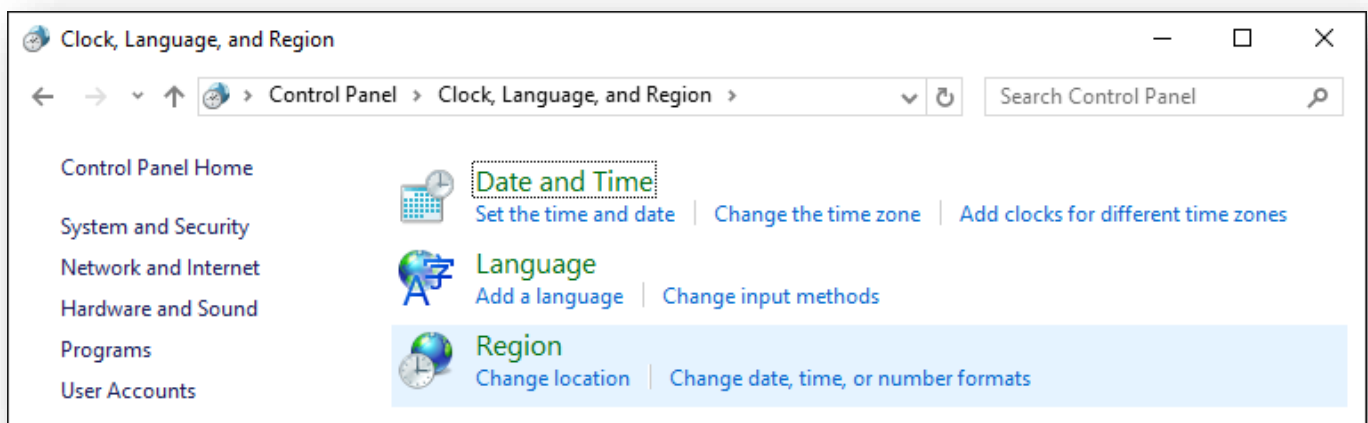
### Примерен вход и изход

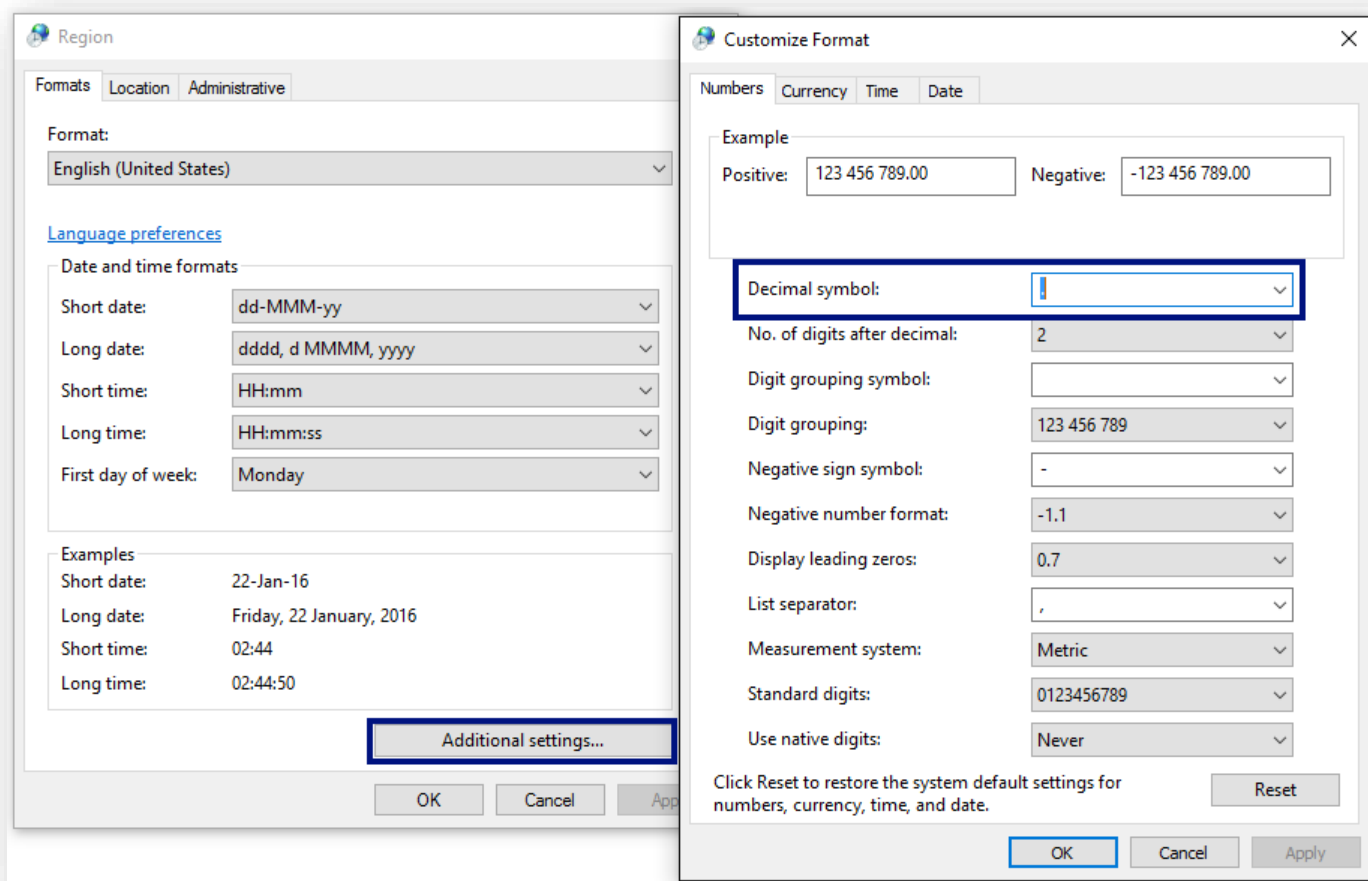
ВХОД	ИЗХОД
5	12.7

ВХОД	ИЗХОД
7	17.78

**Внимание:** в зависимост от регионалните настройки на операционната система, е възможно вместо **десетична точка** (US настройки) да се използва **десетична запетая** (BG настройки). Ако програмата очаква десетична точка и бъде въведено число с десетична запетая или на обратно (бъде въведена десетична точка когато се очаква десетична запетая), то тя няма да може да се изпълни.

Препоръчително е **да промените настройките на компютъра си**, така че да се използва **десетична точка**:



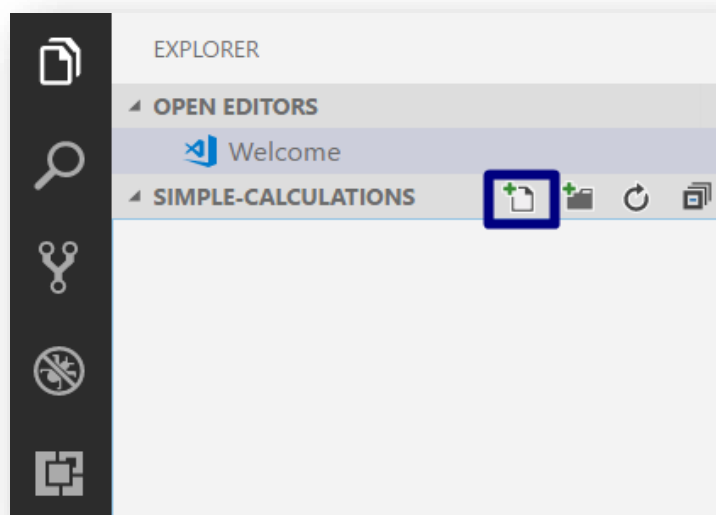


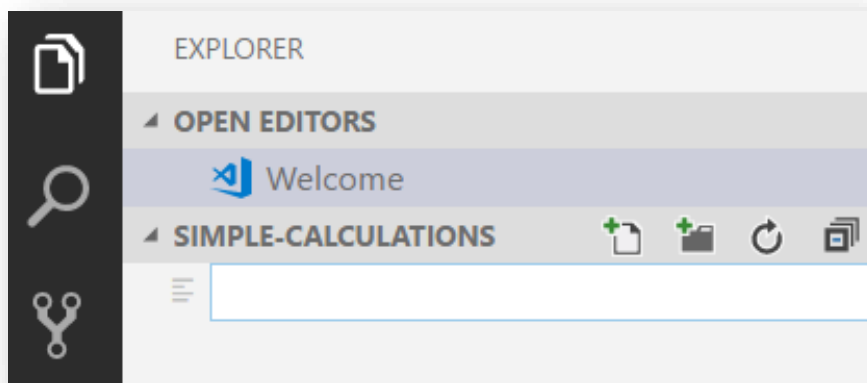
## 5. Поздрав по име

Напишете функция, която **получава име на човек** и отпечатва **"Hello, <name>!"**, където **<name>** е въведеното име от конзолата.

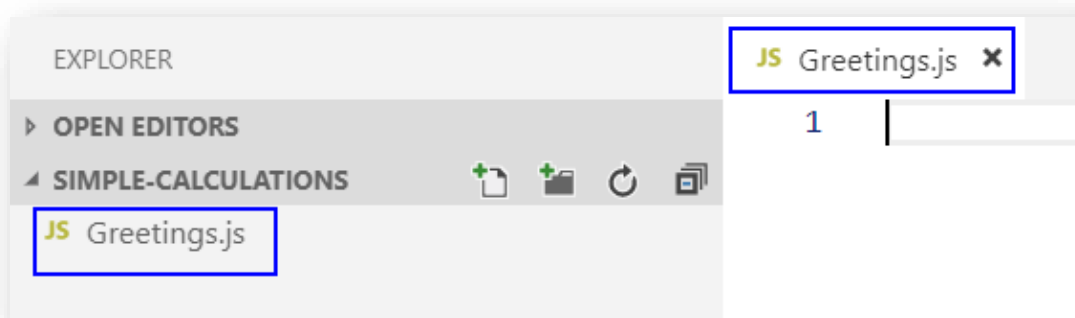
### Насоки

1. Първо създайте **нов JavaScript файл** в съществуващата папка и го именувайте подходящо. Препоръчително е всеки скриптов файл да се казва, както името на задачата, чието решение съдържа.





2. Съдържанието на новият файл ще се отвори в прозореца в дясно



Създайте променливата **name** и запазете в нея името, което се подава от (**input**).

Изведете изхода на конзолата, като използвате следния шаблон:

```
console.log(`Hello, ${name}!`);
```

Как работи примерът? Методът **console.log()** ни позволява в кръглите скоби да записваме поредица от символи, които държат в себе си стойността на дадена променлива – **\${name}** и я принтира на конзолата.

Забележете, че за да работи текстът трябва да е ограден от символа **`**, който разпознава обикновен текст и променлива. За да бъде разпозната една променлива, като такава, тя трябва да започва със символа **\$** и да бъде оградена в кърдрави скоби : **{ }**.

3. Стартирайте програмата с **Ctrl + F5** и тествайте с различни входни примери.
4. За да получим резултат трябва да "повикаме" функцията и да ѝ подадем входни данни:



```
function greetings(name) {  
    console.log(`Hello, ${name}!`)  
}  
  
greetings("Niki");
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL

C:\Program Files\nodejs\node.exe Greetings.js  
Hello, Niki!

5. Тествайте решението си в judge, без да включвате извикването на функцията .

## 6. Съединяване на текст и числа

Напишете функция, която **получава: име, фамилия, възраст и град** и печата съобщение от следния вид:

"You are <firstName> <lastName>, a <age>-years old person from <town>."

### Насоки

1. Добавете към текущото Visual Studio Code решение още един **JavaScript файл** с име "**Concatenate-Data**".
2. Приемете от конзолата входните данни:

```
function concatenateData(firstName, lastName, age, town) {  
    // Your code here  
}  
  
concatenateData("Maria", "Ivanova", 20, "Sofia");
```

3. Изведете на конзолата форматирания изход.
4. Стартирайте програмата с **Ctrl + F5** и тествайте с различни входни примери. За да изведете локално резултат на конзолата, трябва да извикате функцията и да подадете входни данни в реда, в който очаквате да ги получите:

```
C:\Program Files\nodejs\node.exe Concatenate-Data.js  
You are Maria Ivanova, a 20-years old person from Sofia.
```

## 7. Изготвяне на проекти

Напишете програма, която **изчислява колко часа** ще са необходими на един архитект, за да **изготви проектите** на няколко строителни обекта. Изготвянето на един проект отнема **три часа**.

### Вход

От конзолата се четат **2 реда**:

1. Името на архитекта - текст
2. Брой на проектите, които трябва да изготви - цяло число в интервала [0 ... 100]

### Изход

На конзолата се отпечатва:

- "The architect {името на архитекта} will need {необходими часове} hours to complete {брой на проектите} project/s."

### Примерен вход и изход

Вход	Изход	Вход	Изход
"George", 4	The architect George will need 12 hours to complete 4 project/s.	"Sanya", 9	The architect Sanya will need 27 hours to complete 9 project/s.

## 8. Зоомагазин

Напишете програма, която **пресмята нужните разходи** за закупуването на храна за кучета и котки. Храната се пазарува от зоомагазин, като една опаковка храна за **кучета е на цена 2.50 лв**, а опаковка храна за **котки струва 4 лв**.

### Вход

От конзолата се четат **2 реда**:

1. Броят на опаковките храна за кучета – цяло число в интервала [0... 100]
2. Броят на опаковките храна за котки – цяло число в интервала [0... 100]

### Изход

На конзолата се отпечатва:

"{крайната сума} lv."

### Примерен вход и изход

Вход	Изход	Вход	Изход
5, 4	28.5 lv.	13, 9	68.5 lv.

## 9. Озеленяване на дворове

Божидара разполага с **няколко къщи** на Черноморието и **желае да озелени дворовете на някои от тях**, като по този начин създаде **уютна обстановка и комфорт на гостите си**. За целта е наела фирма.

Напишете програма, която **изчислява необходимата сума**, които Божидара ще трябва да заплати на фирмата изпълнител на проекта. Цената на **един кв. м. е 7.61 лв. с ДДС**. Понеже нейният двор е **доста голям**, фирмата изпълнител предлага **18% отстъпка от крайната цена**.

### Вход

От конзолата се прочита само **един ред**:

1. Кв. метри, които ще бъдат озеленени – реално число в интервала [0.00 ... 10000.00]

### Изход

На конзолата се отпечатват **два реда**:

- "The final price is: {крайна цена на услугата} lv."
- "The discount is: {отстъпка} lv."

### Примерен вход и изход

Вход	Изход	Обяснения
550	The final price is: 3432.11 lv. The discount is: 753.39 lv.	Пресмятаме цената за озеленяване на целия двор: $550 * 7.61 = 4185.50$ лв. Приспадаме отстъпката ( $18\% = 0.18$ ) от общата сума: $0.18 * 4185.5 = 753.39$ лв. Калкулираме крайната цена на услугата: $4185.50 - 753.39 \rightarrow 3432.11$ лв.
Вход	Изход	
150	The final price is: 936.03 lv. The discount is: 205.47 lv.	Пресмятаме цената за озеленяване на целия двор: $150 * 7.61 = 1141.50$ лв. Приспадаме отстъпката ( $18\% = 0.18$ ) от общата сума: $0.18 * 1141.50 = 205.47$ лв. Калкулираме крайната цена на услугата: $1141.50 - 205.47 \rightarrow 936.03$ лв.