

Първи стъпки в програмирането

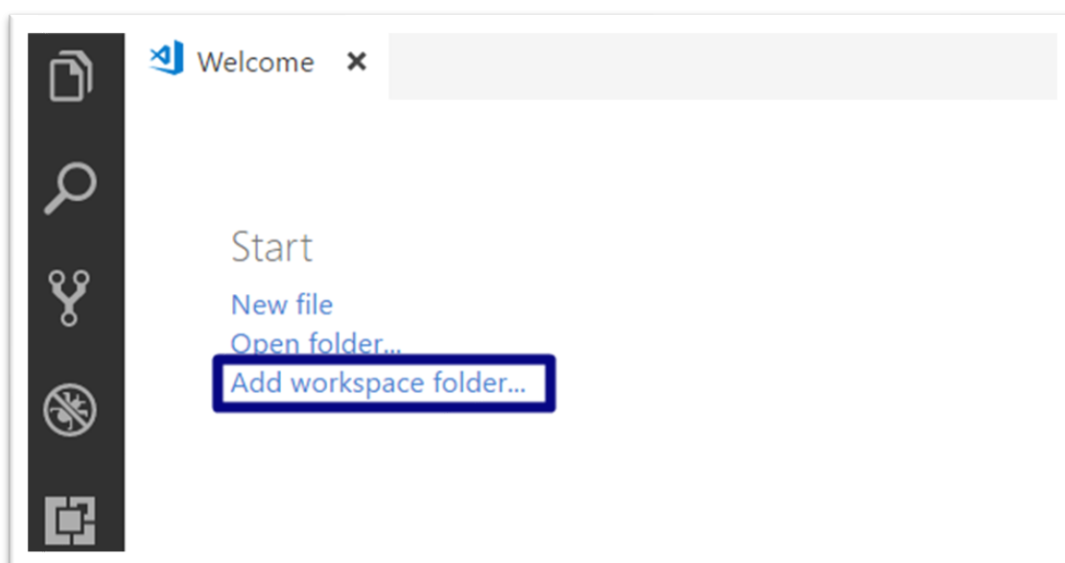
Задачи за упражнение в клас и за домашно към курса "[Основи на програмирането](#)" @ СофтУни.

Тествайте решенията си в онлайн **judge системата**: <https://judge.softuni.bg/Contests/Compete/Index/2399#0>

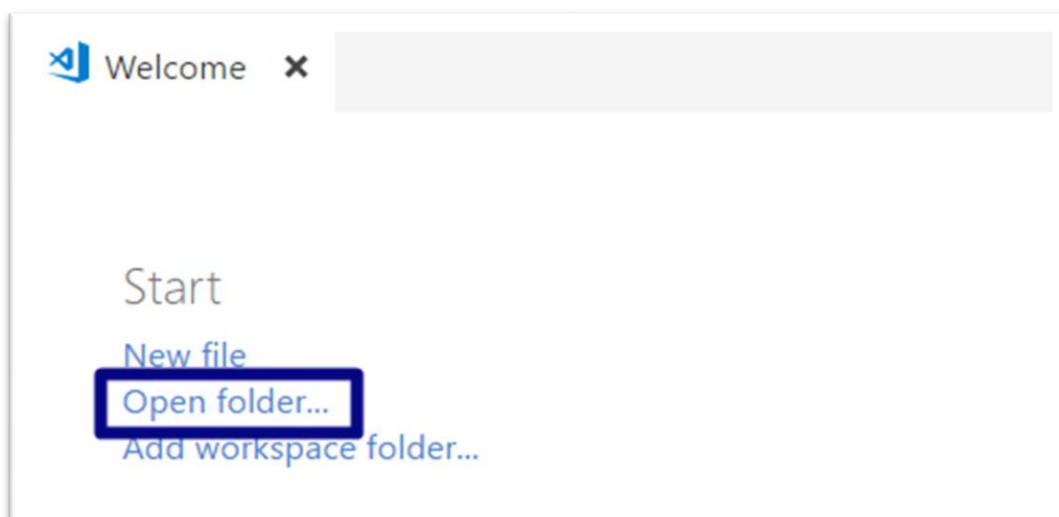
Празен Visual Studio Code проект

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.

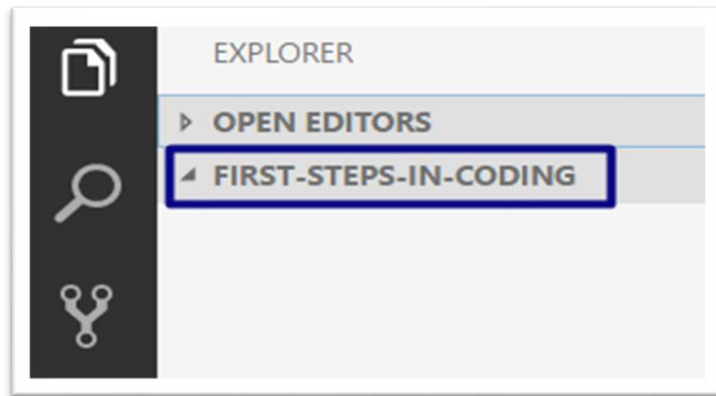
1. Стартирайте Visual Studio Code
2. Създайте нова папка, която ще държи отделните решения. Ще се отвори диалогов прозорец, в който ще трябва да изберете нейната директория. Препоръчително е да именувате папката според темата на заданието, пример "**First-Steps-in-Calculations**"



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



Панелът в ляво ще изглежда така:



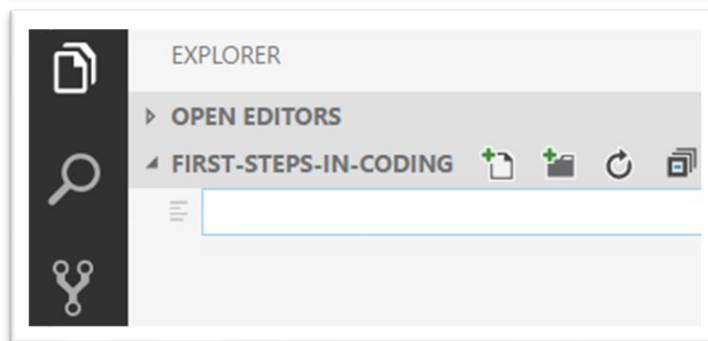
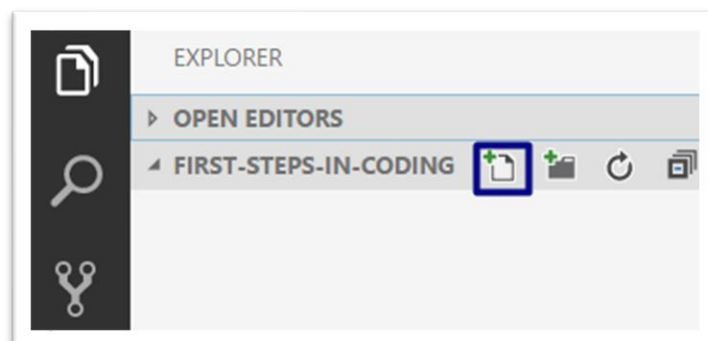
1. Конзолна програма "Hello SoftUni"

Първата задача от тази тема е да се напише **функция**, която **отпечатва** на конзолата: "Hello SoftUni".

Примерен вход и изход

ВХОД	ИЗХОД
(няма)	Hello SoftUni

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 "повикаме" функцията:

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

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

5. Тествайте решението на тази задача в онлайн judge системата на СофтУни:

01. Hello SoftUni

[Participants](#)[Tests](#)[Change](#)[Delete](#)[Administration](#)

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

Allowed working time: 0.100 sec.

Allowed memory: 16.00 MB

Size limit: 16.00 KB

Checker: Trim ?

JavaScript code (Nod... ▾

[Submit](#)

Submissions



Points	Time and memory used	Submission date
✓ 100 / 100	Memory: 11.14 MB Time: 0.140 s	22:51:56 15.02.2018

[Details](#)

2. Числата от 1 до 10

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

Примерен вход и изход

вход	изход
(няма)	1 2 3 4 5 6 7 8 9 10

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.

```
JS Nums-1-To-10.js
JS Nums-1-To-10.js > nums1to20
1 function nums1to20() {
2     console.log(1);
3     console.log(2);
4     console.log(3);
5     console.log(4);
6     console.log(5);
7     console.log(6);
8     console.log(7);
9     console.log(8);
10    console.log(9);
11    console.log(10);
12 }
```

3. Стартирайте програмата с `Ctrl + [F5]`. За да получите резултат трябва да "повикате" функцията.
4. Можете ли да напишете програмата по **по-умен начин**, така че да не повтаряте 10 пъти една и съща команда? Потърсете в Интернет информация за [for loop JavaScript](#).

3. Пресмятане на лице на квадрат

Да се напише **функция**, която **получава цяло число** и **пресмята лицето на квадрат със страна "а"**. Задачата е тривиално лесна: получавате число в аргумента на функцията, умножавате го само по себе си и печатате получения резултат на конзолата.

Примерен вход и изход

вход	изход
(["5"])	25

Насоки

1. Приемете вход (число във вид на стринг) и го запазете в променливата **а**, като го преобразувате към числен тип.
2. Инициализирайте променлива, която държи пресметнатата стойност на лицето, получена по формулата **а * а** и принтирайте резултата, като не забравяйте, че за да получите локално резултат, трябва да извикате функцията и да й подадете входни данни.

```
function squareArea(input) {
  let a = Number(input[0]);
  let area = a * a;
  console.log(area);
}
```

4. От инчове към сантиметри

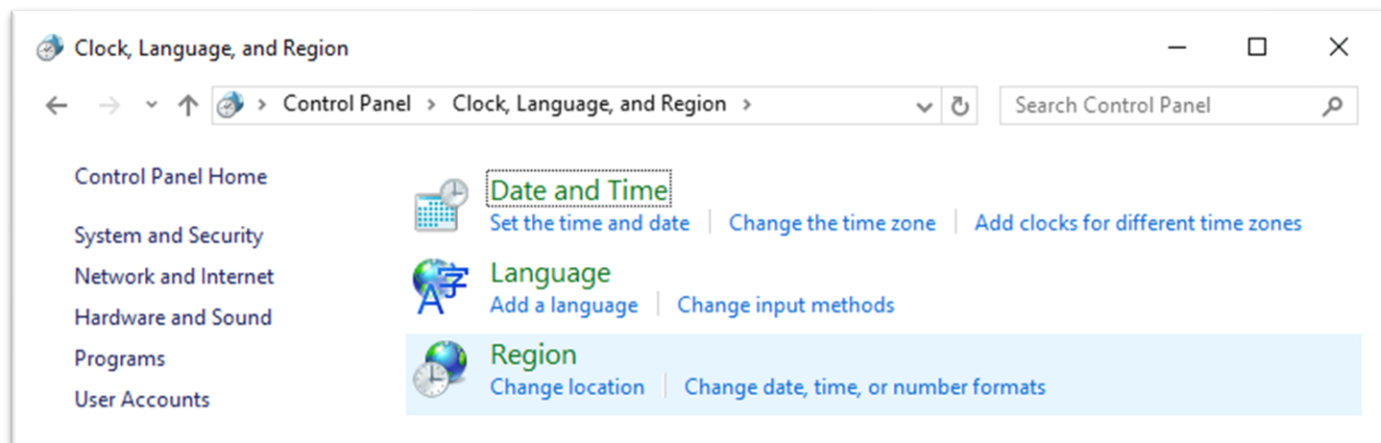
Да се напише функция, която **получава цяло число** и преобразува числото **от инчове в сантиметри**. За целта **умножава инчовете по 2.54** (защото 1 инч = 2.54 сантиметра).

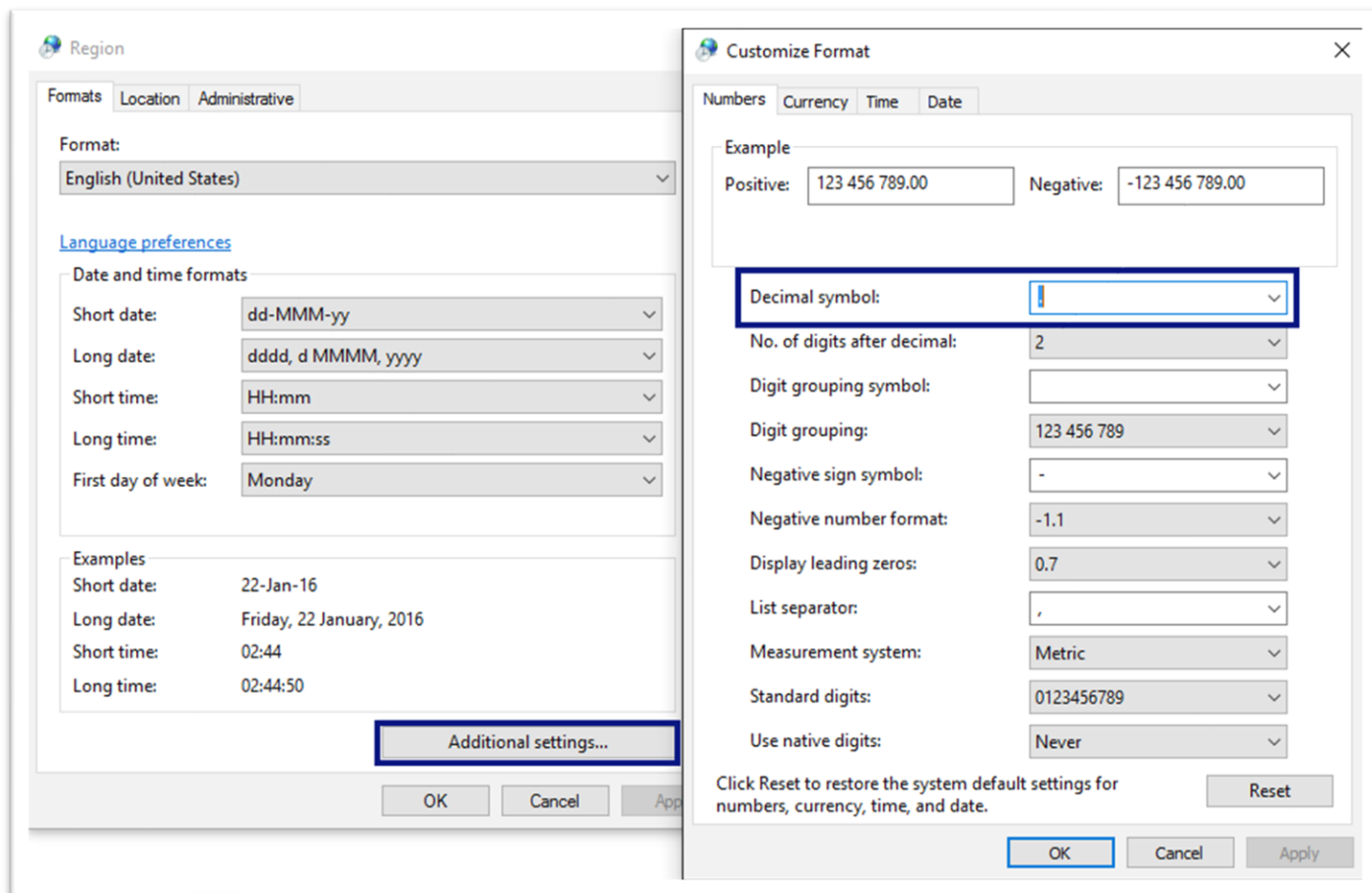
Примерен вход и изход

вход	Изход
(["5"])	12.7

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**:



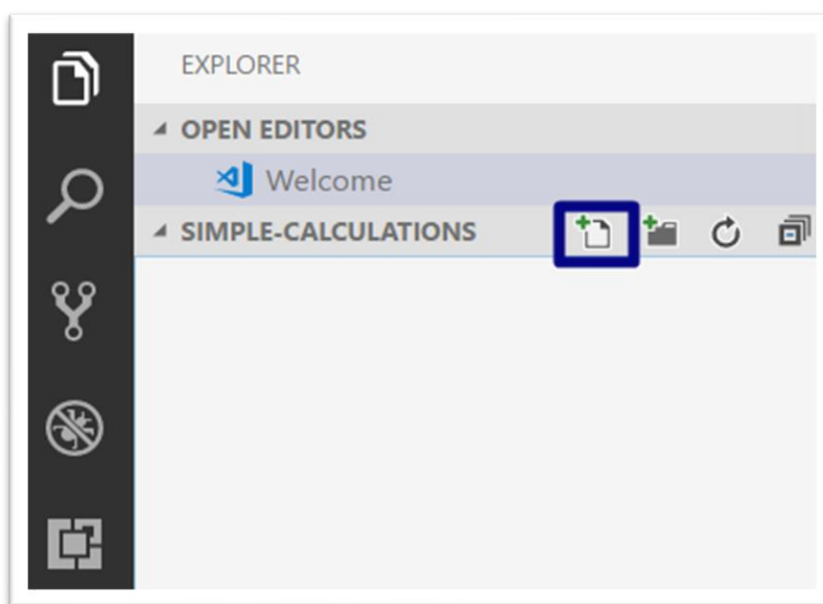


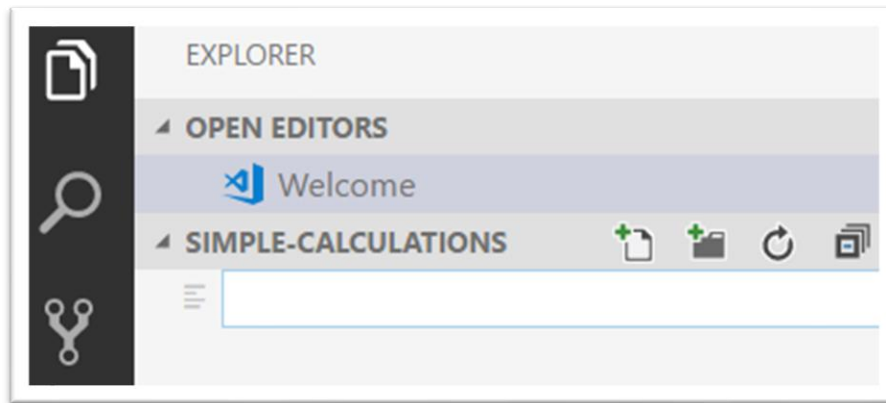
5. Поздрав по име

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

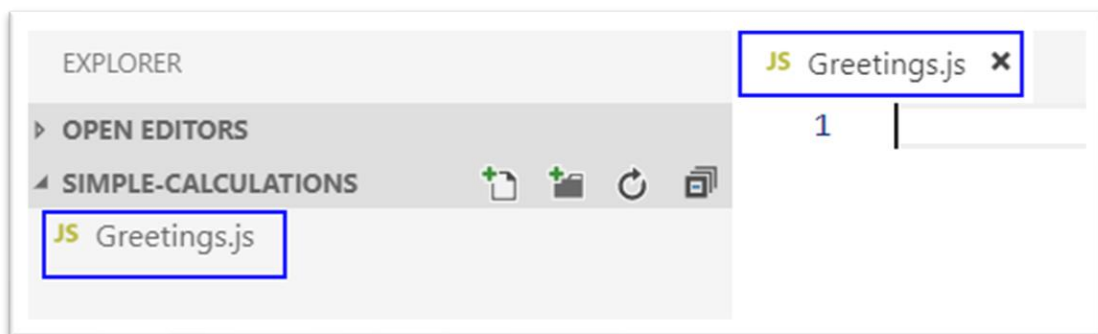
Насоки

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 къдрави 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(input) {  
    let name = input[0];  
    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(input) {  
  
}
```

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:

```
concatenateData(['Maria', 'Ivanova', 20, 'Sofia'])
```

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

Вход

Функцията получава 2 аргумента:

1. Името на архитекта - текст
2. Брой на проектите - цяло число в интервала [0... 100]

Изход

На конзолата се отпечатва:

- "The architect {името на архитекта} will need {необходимите 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 функция, която **пресмята нужните разходи** за закупуването на храна за кучета. Храната се пазарува **основно за кучета**, от зоомагазин, но понякога стопанинът им купува и за **животните на съседа му**. Една опаковка храна за **кучета е на цена 2.50лв.**, а всяка останала, която **не е за тях струва 4лв.**

Вход

Функцията получава 2 аргумента:

1. Броят на кучетата – цяло число в интервала [0... 100]
2. Броят на останалите животни - 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**, като за целта е наела фирма.

Напишете функция, която **изчислява необходимите средства**, които Божидара ще трябва да заплати на фирмата изпълнител на проекта. Цената на **един кв. м. е 7.61лв със ДДС**. Тъй като нейният двор е **доста голям**, фирмата изпълнител предлага **18% отстъпка от крайната цена**.

Вход

Функцията получава един аргумент:

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: $550 * 7.61 = 4185.5$ лв. Приспадаме отстъпката от общата сума: $0.18 * 4185.5 = 753.39$ лв. Калкулираме крайната цена на услугата: $4185.5 - 753.39 \rightarrow 3432.11$ лв.
Вход	Изход	
(["150"])	The final price is: 936.03 lv. The discount is: 205.47 lv.	