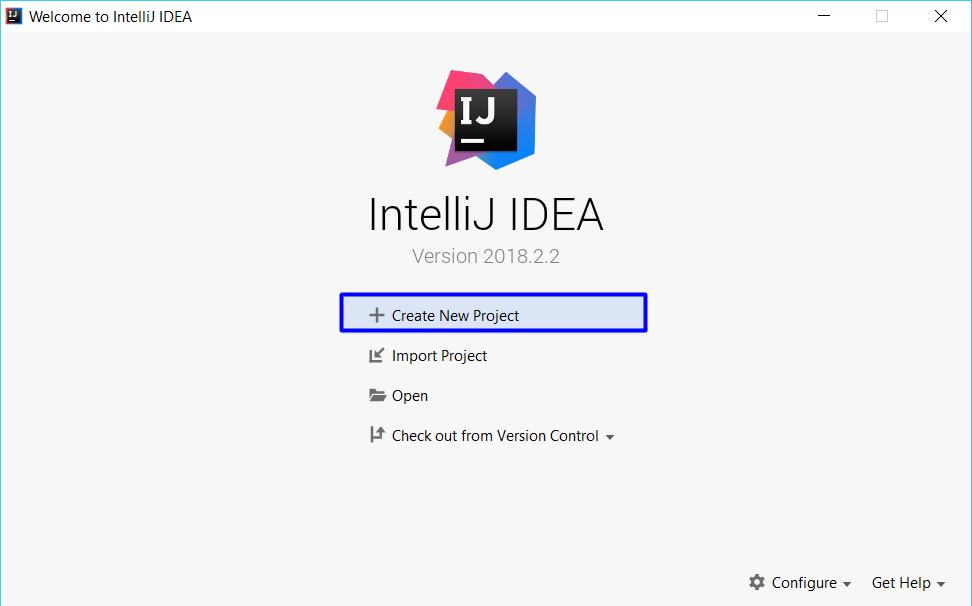
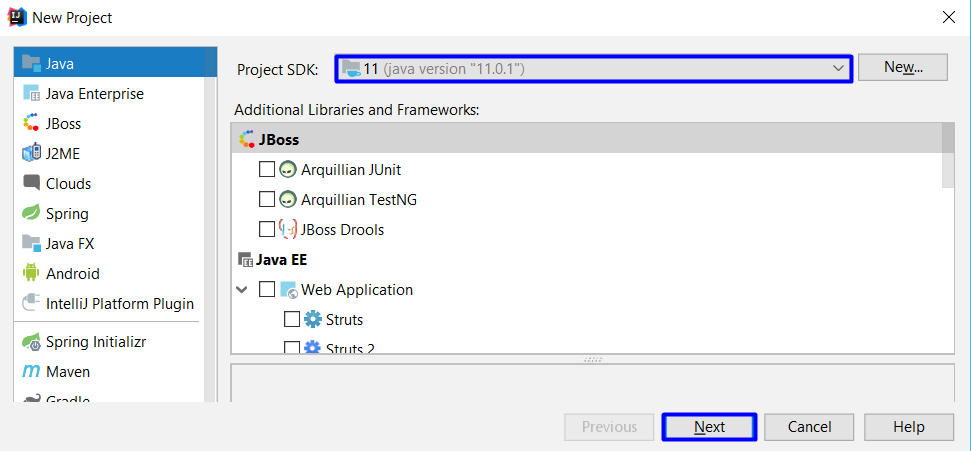
# Лаб: Вложени условни конструкции

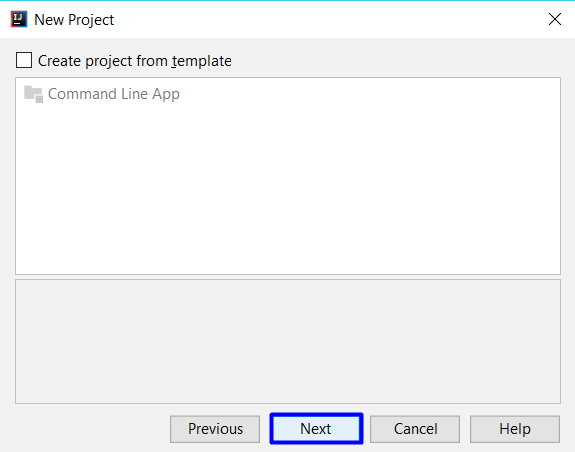
Задачи за упражнение в клас и за домашно към курса ["Основи на програмирането" @ СофтУни](https://softuni.bg/courses/programming-basics).

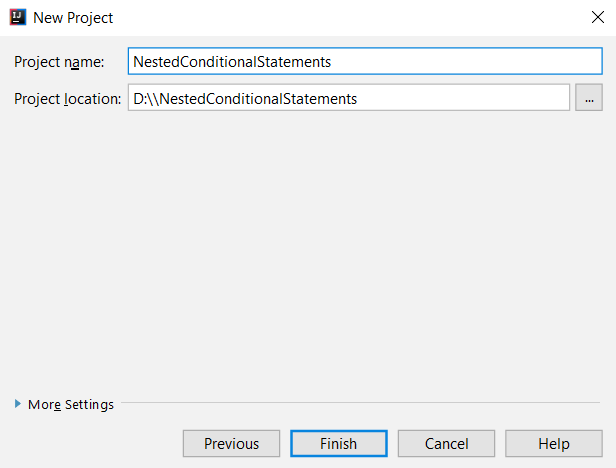
**Тествайте** решението си в **judge системата**: <https://judge.softuni.bg/Contests/Compete/Index/1013#0>

## Създаване на нов проект

Създайте нов проект в IntelliJ IDEA







## Обръщение според възраст и пол

Да се напише **конзолна програма**, която **прочита възраст** (десетично число) и **пол** ("m" или "f"), въведени от потребителя, и отпечатва **обръщение** измежду следните:

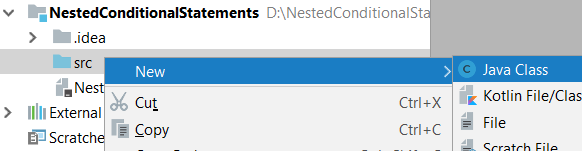
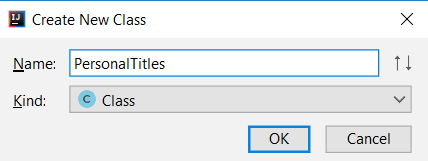
* "Mr." - мъж (пол "m") на 16 или повече години
* "Master" - момче (пол "m") под 16 години
* "**Ms.**" - жена (пол "f") на 16 или повече години
* "**Miss**" - момиче (пол "f") под 16 години

### Примерен вход и изход

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 12  f | Miss | 17  m | Mr. | 25  f | Ms. | 13.5  m | Master |

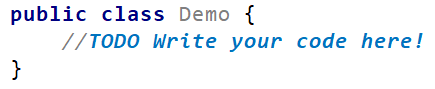
### Насоки

1. Създайте **нов проект** в съществуващото IntelliJ решение. Кликнете с десен бутон на мишката върху **папката 'src'**. Изберете [New] 🡪 [Class]:

 🡺 

Вече имате проект с едно конзолно приложение в него. Остава да напишете кода за решаване на задачата.

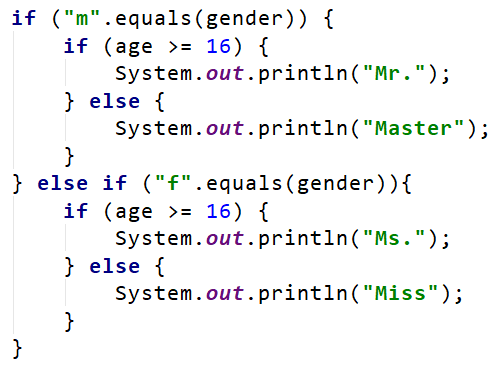
1. Създайте main метод и напишете решението на задачата. Можете да си помогнете с кода от картинките по-долу:



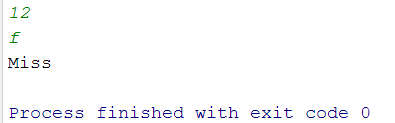
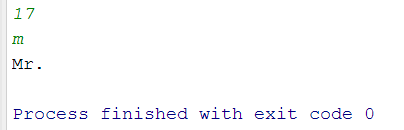
1. Прочетете входните данни от конзолата. Първо прочетете едно реално число, "age", а на следващия ред "String / текст" за "gender".



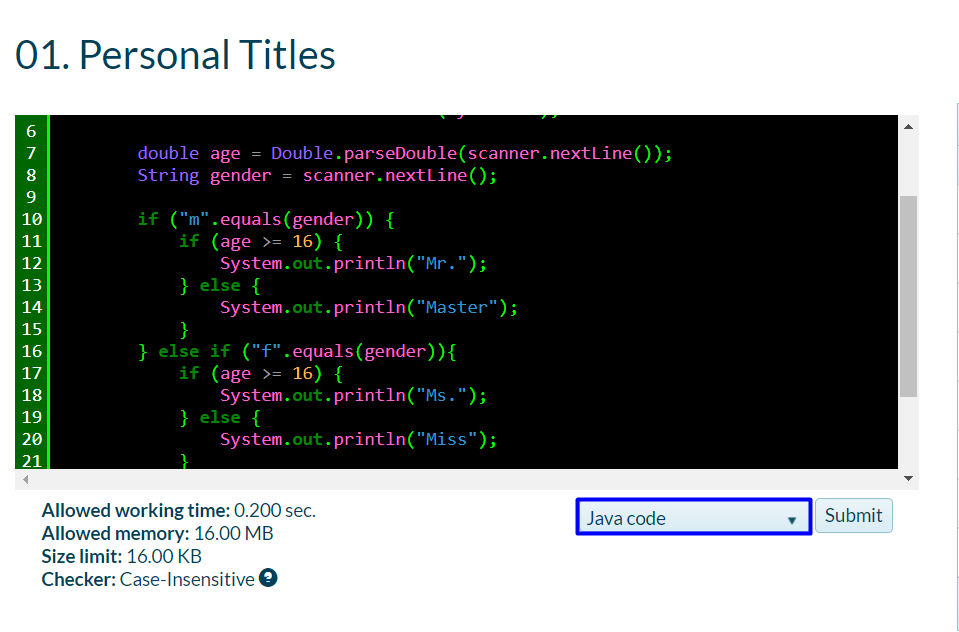
1. Направете проверка за пола, използвайки метода "equals", при резултат "true" направете серия от проверки за възрастта и отпечатайте на конзолата желаното обръщение.



1. **Стартирайте** програмата с [Ctrl+Shift+F10] и я **тествайте** с различни входни стойности:

1. Трябва да получите **100 точки** (напълно коректно решение):



## Квартално магазинче

Предприемчив българин отваря **квартални магазинчета** в **няколко града** и продава на **различни цени**:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| град / продукт | **coffee** | **water** | **beer** | **sweets** | **peanuts** |
| **Sofia** | 0.50 | 0.80 | 1.20 | 1.45 | 1.60 |
| **Plovdiv** | 0.40 | 0.70 | 1.15 | 1.30 | 1.50 |
| **Varna** | 0.45 | 0.70 | 1.10 | 1.35 | 1.55 |

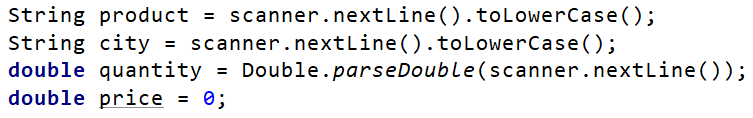
Напишете програма, която чете **продукт** (низ), **град** (низ) и **количество** (десетично число), въведени от потребителя, и пресмята и отпечатва **колко струва** съответното количество от избрания продукт в посочения град.

### Примерен вход и изход

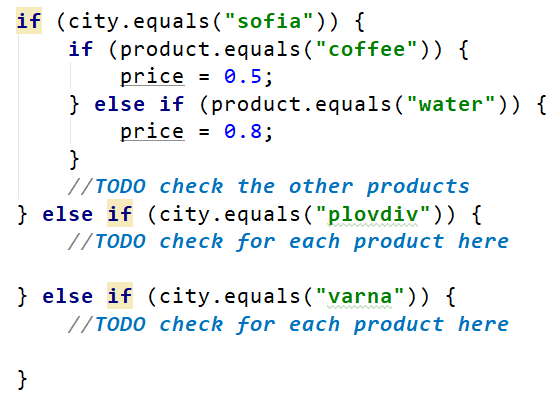
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| coffee  Varna  2 | 0.9 | peanuts  Plovdiv  1 | 1.5 | beer  Sofia  6 | 7.2 | water  Plovdiv  3 | 2.1 | sweets  Sofia  2.23 | 3.2335 |

### Насоки

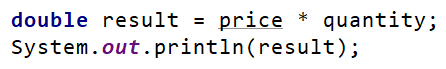
1. Първо създайте **нова Java програма**
2. Отидете в тялото на метода main(String[] args) и напишете решението на задачата. Можете да си помогнете с кода от картинката по-долу. Можете да прехвърлите всички букви в долен регистър с .toLowerCase() за да сравнявате продукти и градове без значение на малки / главни букви:
3. Прочетете входните данни от конзолата и си създайте една променлива price, като и зададете стойност "0".



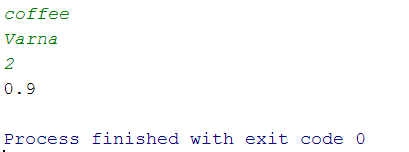
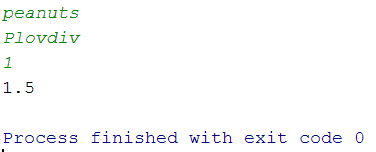
1. Направете серия от проверки, като за всеки град проверете за дадения продукт. Във всяка проверка за продукт, променяйте стойността на променливата priceи я принтирайте. Вижте примера по-долу.



1. Създайте променлива "result" от тип "double", в която да пресметнете и запазите крайния резултат. Отпечатайте "result" на конзолата.



**Стартирайте** програмата с [Ctrl+Shift+F10] и я **тествайте** с различни входни стойности:

## Точка в правоъгълник

Напишете програма, която проверява дали **точка {x, y}** се намира **вътре в правоъгълник {x1, y1} – {x2, y2}**. Входните данни се четат от конзолата и се състоят от 6 реда, въведени от потребителя: десетичните числа **x1**, **y1**, **x2**, **y2**, **x** и **y** (като се гарантира, че **x1 < x2** и **y1 < y2**). Една точка е вътрешна за даден правоъгълник, ако се намира някъде във вътрешността му или върху някоя от страните му. Отпечатайте "**Inside**" или "**Outside**".

### Примерен вход и изход

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** | **визуализация** |  | **вход** | **изход** | **визуализация** |
| 2  -3  12  3  8  -1 | Inside |  | 2  -3  12  3  11  -3.5 | Outside |  |

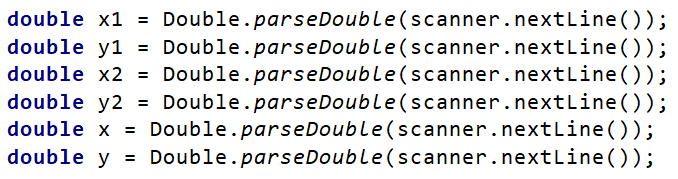
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** | **визуализация** |  | **вход** | **изход** | **визуализация** |
| -1  -3  4  1  0.5  1 | Inside |  | -1  -3  4  1  -1.2  1.4 | Outside |  |

\* **Подсказка**: една точка е вътрешна за даден многоъгълник, ако едновременно са изпълнени следните четири условия (можете да ги проверите с if проверка с логическо "**и**" - оператор &&):

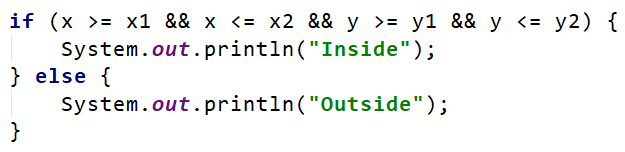
* Точката е надясно от лявата стена на правоъгълника (**x >= x1**)
* Точката е наляво от дясната стена на правоъгълника (**x <= x2**)
* Точката е надолу от горната стена на правоъгълника (**y >= y1**)
* Точката е нагоре от долната стена на правоъгълника (**y <= y2**)

### Насоки

1. Прочетете входните данни от конзолата.



1. Направете проверка на горепосочените условия използвайки if проверка с логическо "и" - оператор && и в тялото на проверката принтирайте "Inside", ако проверката е с резултат "false", принтирайте "Outside".



## Плод или зеленчук?

Да се напише програма, която **чете име на продукт**, въведено от потребителя, и проверява дали е **плод** или **зеленчук**.

* Плодовете "**fruit**" са **banana**, **apple**, **kiwi**, **cherry**, **lemon** и **grapes**
* Зеленчуците "**vegetable**" са **tomato**, **cucumber**, **pepper** и **carrot**
* Всички останали са "**unknown**"

Да се изведе "**fruit**", "**vegetable**" или "**unknown**" според въведения продукт.

### Примерен вход и изход

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| banana | fruit | apple | fruit | tomato | vegetable | water | unknown |

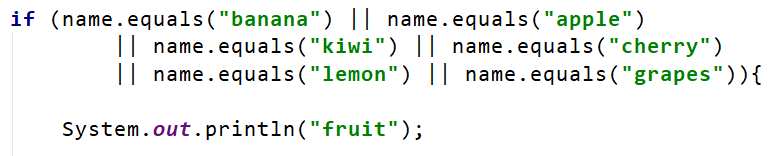
\* **Подсказка**: използвайте условна if проверка с логическо "**или**" - operator ||.

### Насоки

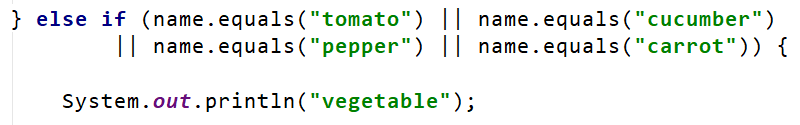
1. Прочетете входните данни от конзолата.



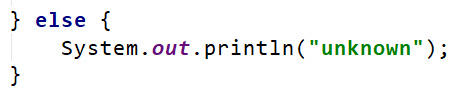
1. Проверете името на продукта дали съвпада с имената на плодове с помощта на логическо "или" - оператор "||" и при резултат "true", отпечатайте на конзолата "fruit".



1. Ако горепосочената проверка върне "false" проверете дали името на продукта не съвпада с някое от имената на зеленчуците отново с помощта на логическо "или" - оператор "||", при "true" отпечатайте на конзолата "vegetable".



1. При невалидно име на продукта, принтирайте на конзолата "unknown".



## Невалидно число

Дадено **число е валидно**, ако е в диапазона [**100**…**200**] или е **0**. Да се напише програма, която **чете цяло число**, въведено от потребителя, и печата "**invalid**" ако въведеното число **не е валидно**.

### Примерен вход и изход

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 75 | invalid | 150 | *(няма изход)* | 220 | invalid | 199 | *(няма изход)* |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| -1 | invalid | 100 | *(няма изход)* | 200 | *(няма изход)* | 0 | *(няма изход)* |

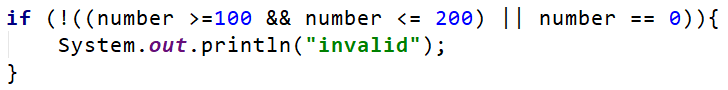
\* **Подсказка**: използвайте условна if проверка с **отрицание** и логически операции.

### Насоки

1. Прочетете цяло число от конзолата.



1. Направете проверка дали дадено число **не** е валидно като в if проверка използвате логически оператор за отрицание "!" и проверите дали даденото число "number" е по-голямо или равно на 100 и същевременно е по-малко или равно на 200 "number >= 100 && number <= 200", или числото е равно на 0, "number == 0". Ако проверката върне "true", принтирайте на конзолата "invalid".



## Магазин за плодове

Магазин за плодове през **работните дни** работи на следните **цени**:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **плод** | **banana** | **apple** | **orange** | **grapefruit** | **kiwi** | **pineapple** | **grapes** |
| **цена** | 2.50 | 1.20 | 0.85 | 1.45 | 2.70 | 5.50 | 3.85 |

**Събота** и **неделя** магазинът работи на **по-високи** **цени**:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **плод** | **banana** | **apple** | **orange** | **grapefruit** | **kiwi** | **pineapple** | **grapes** |
| **цена** | 2.70 | 1.25 | 0.90 | 1.60 | 3.00 | 5.60 | 4.20 |

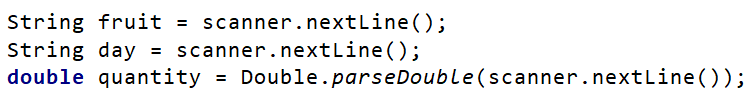
Напишете програма, която чете от конзолата **плод** (banana / apple / orange / grapefruit / kiwi / pineapple / grapes), **ден от седмицата** (Monday / Tuesday / Wednesday / Thursday / Friday / Saturday / Sunday) и **количество** (десетично число) , въведени от потребителя, и пресмята **цената** според цените от таблиците по-горе. При невалиден ден от седмицата или невалидно име на плод да се отпечата "error".

### Примерен вход и изход

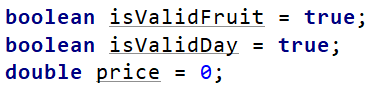
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| apple  Tuesday  2 | 2.40 | orange  Sunday  3 | 2.70 | kiwi  Monday  2.5 | 6.75 | grapes  Saturday  0.5 | 2.10 | tomato  Monday  0.5 | error |

### Насоки

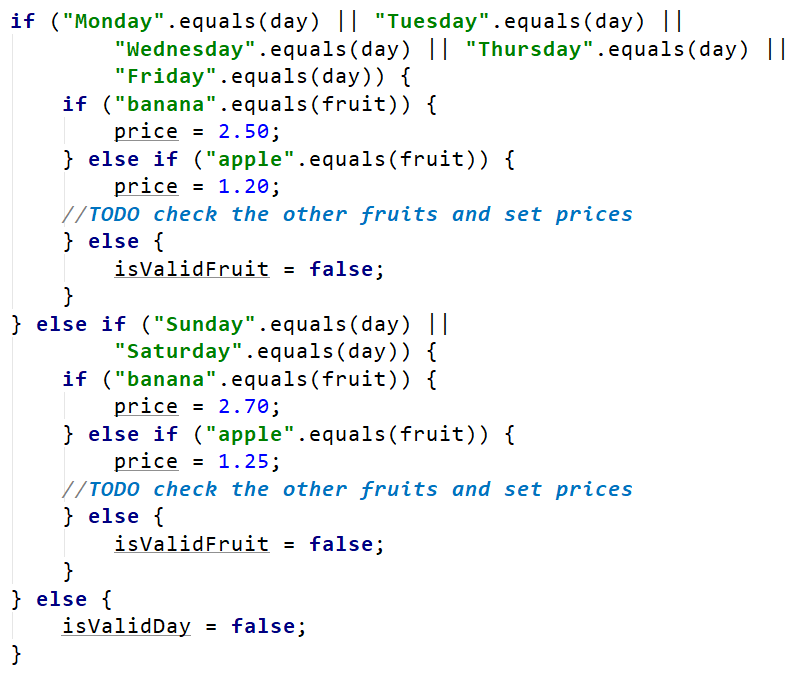
1. Прочетете входните данни от конзолата.



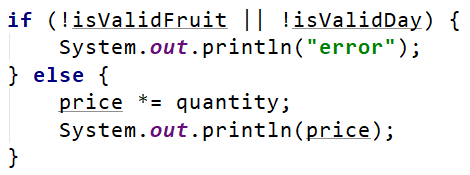
1. Създайте три променливи както следва, две от тип "boolean", "isValidFruit" и "isValidDay" с първоначална стойност "true", и една променлива тип "double", "price" с първоначална стойност "0".



1. В серия от проверки проверете за деня от седмицата използвайки логическо "или" – оператор "||" и за всяка проверка за дните, проверявайте плода, като за всеки плод променяйте цената "price". При невалиден ден променете стойността на променливата "isValidDay" на "false", a когато плодът е невалиден променете стойността на променливата "isValidFruit" отново на "false"



1. Проверете стойностите на булевите променливи чрез логическия оператор "или"-"||" и, ако **едната или двете** са със стойност "false", принтирайте на конзолата "error", в противен случай принтирайте единичната цена "price" умножена по количеството, "quantity".



## Търговски комисионни

Фирма дава следните **комисионни** на търговците си според **града**, в който работят и обема на **продажбите** s:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Град** | **0 ≤ s ≤ 500** | **500 < s ≤ 1 000** | **1 000 < s ≤ 10 000** | **s > 10 000** |
| Sofia | 5% | 7% | 8% | 12% |
| Varna | 4.5% | 7.5% | 10% | 13% |
| Plovdiv | 5.5% | 8% | 12% | 14.5% |

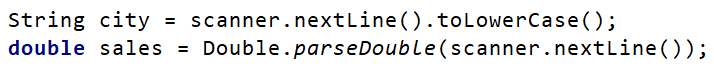
Напишете **конзолна програма**, която чете име на **град** (стринг) и обем на **продажби** (десетично число) , въведени от потребителя, и изчислява и извежда размера на търговската **комисионна** според горната таблица. Резултатът да се изведе форматиран до **2 цифри след десетичната точка**. При **невалиден** град или обем на продажбите (отрицателно число) да се отпечата "**error**".

### Примерен вход и изход

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| Sofia  1500 | 120.00 | Plovdiv  499.99 | 27.50 | Varna  3874.50 | 387.45 | Kaspichan  -50 | error |

### Насоки

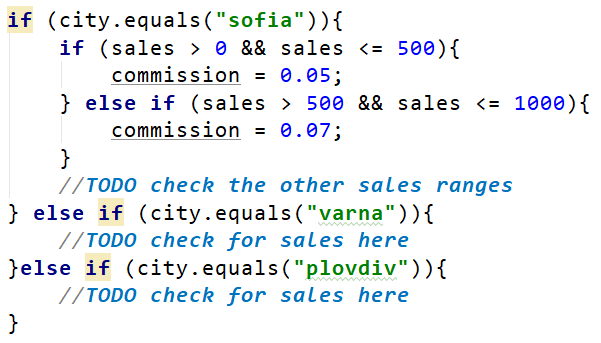
1. Прочетете входните данни от конзолата, **обърнете града в** **малки букви**,чрез метода .toLowerCase().



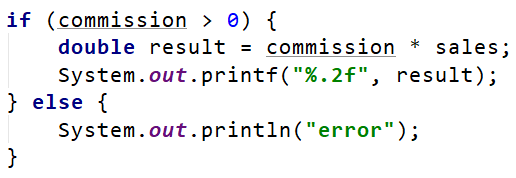
1. Първоначално задайте **комисионна -1**. Тя ще бъде променена, ако градът и ценовият диапазон бъдат намерени в таблицата с комисионните.



1. Използвайте вложени if проверки, за **да изчислите комисионната** според града и според обема на продажбите. Може да си помогнете с кода по-долу:



1. Направете проверка на комисионната, ако е **по голяма** от "0", създайте нова променлива от тип "double" – "result", като в нея запазвате крайния резултат изчислявайки (комисионна по обем на продажбите) и принтирайте на конзолата. В противен случай принтирайте "error".



**Примерна изпитна задача**

## \*Ски почивка

Атанас решава да прекара отпуската си в Банско и да кара ски. Преди да отиде обаче, трябва да резервира хотел и да изчисли **колко ще му струва престоя**. Съществуват следните видове помещения, със следните цени за престой:

* + - **"room for one person" – 18.00 лв за нощувка**
    - "**apartment**" – **25.00 лв за нощувка**
    - **"president** **apartment**" – **35.00 лв за нощувка**

Според **броят на дните**, в които ще остане в хотела (**пример: 11 дни = 10 нощувки**) и **видът на помещението**, което ще избере, той може да ползва различно **намаление**. Намаленията са както следва:

|  |  |  |  |
| --- | --- | --- | --- |
| **вид помещение** | **по-малко от 10 дни** | **между 10 и 15 дни** | **повече от 15 дни** |
| **room for one person** | не ползва намаление | не ползва намаление | не ползва намаление |
| **apartment** | **30%** от крайната цена | **35%** от крайната цена | **50%** от крайната цена |
| **president apartment** | **10%** от крайната цена | **15%** от крайната цена | **20%** от крайната цена |

След престоя, оценката на Атанас за услугите на хотела може да е **позитивна (positive)** или **негативна (negative)** . Ако оценката му е **позитивна**, към цената **с вече приспаднатото намаление** Атанас добавя **25%** от нея. Ако оценката му е **негативна** приспада от цената **10%**.

### Вход

Входът се чете от конзолата и се състои от **три реда**:

* **Първи ред** - **дни за престой** - **цяло число** в интервала **[0...365]**
* **Втори ред** - **вид помещение -** "room for one person", "apartment" или "**president** apartment"
* **Трети ред** - **оценка** - "positive" или "negative"

### Изход

На конзолата трябва да се отпечата **един ред**:

* **Цената за престоят му в хотела, форматирана до втория знак след десетичната запетая.**

### Примерен вход и изход

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Вход** | **Изход** | **Обяснения** | | | |
| 14  apartment  positive | 264.06 | **14 дни => 13 нощувки** => 13 \* 25.00 = 325 лв.  **10 < 13 дни < 15** => 325 – 35%= 211.25 лв.  **Оценката е positive** => 211.25 + 25% = 264.0625 -> 264.06 лв. | | | |
| **Вход** | **Изход** | **Вход** | **Изход** | **Вход** | **Изход** |
| 30  president apartment  negative | 730.80 | 12  room for one person  positive | 247.50 | 2  apartment  positive | 21.88 |