# Лаб: Условни конструкции

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

Тествайте решенията си в **Judge** системата:

<https://alpha.judge.softuni.org/contests/conditional-statements-lab/2401>

## Отлична оценка

Първата задача от тази тема е да се напише **функция**, която **чете оценка** и отпечатва "**Excellent!",** ако оценката е **5.50** или по-висока.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 6 | Excellent! | 5 | (няма изход) | 5.50 | Excellent! | 5.49 | (няма изход) |

### 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.





1. Съдържанието на новият файл ще се отвори в прозореца вдясно.



1. Отидете във файла **isExcellent.js** и създайте функцията **isExcellent(grade)**.:

Картина, която съдържа Шрифт, текст, линия, бял

Описанието е генерирано автоматично

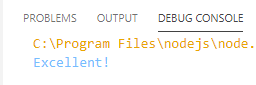
1. Направете проверка за стойността на оценката. Ако тя е по-голяма или равна на 5.50 отпечатайте изхода по условие:

Картина, която съдържа текст, Шрифт, почерк, екранна снимка

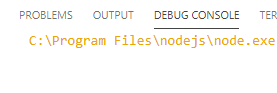
Описанието е генерирано автоматично

1. Извикайте функцията с различни входни стойности и я стартирайте с **Ctrl + F5**:









## По-голямото число

Да се напише функция, която получава **две цели числа** и отпечатва **по-голямото от двете**.

### Примерен вход и изход

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 5, 3 | 5 | 3, 5 | 5 | 10, 10 | 10 | -5, 5 | 5 |

### Насоки

1. Сравнете, дали първото число numOne e по-голямо от второто numTwo.
2. Отпечатайте по-голямото число.

Картина, която съдържа текст, екранна снимка, Шрифт, линия

Описанието е генерирано автоматично

## Четно или нечетно

Да се напише функция, която получава **цяло число** като аргумент и отпечатва на конзолата, дали е **четно** или **нечетно**. Ако е **четно** отпечатайте "**even"**, ако е нечетно "**odd"**.

### Примерен вход и изход

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 2 | even | 3 | odd | 25 | odd | 1024 | even |

### Насоки

1. Първо добавете **нов JavaScript файл**  към съществуващия проект
2. Проверете дали числото е четно, като го разделите на 2 и проверите дали има остатък от делението. Отпечатайте изхода по условие – текста "even" или "**odd**".

Картина, която съдържа текст, Шрифт, почерк, екранна снимка

Описанието е генерирано автоматично

## Познай паролата

Да се напише функция, която **получава парола** (текст) и проверява дали дадената паролата **съвпада** с фразата "s3cr3t!P@ssw0rd".При съвпадение да се изведе "**Welcome**". При несъвпадение да се изведе "**Wrong password!**".

### Примерен вход и изход

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| "qwerty" | Wrong password! | "s3cr3t!P@ssw0rd" | Welcome | "s3cr3t!p@ss" | Wrong password! |

## Число от 100 до 200

Да се напише функция, която **получава цяло число** и проверява дали е **под 100**, **между 100 и 200** или **над 200**. Ако числото е:

* под 100 отпечатайте: **"Less than 100"**
* между 100 и 200 отпечатайте: **"Between 100 and 200"**
* над 200 отпечатайте: **"Greater than 200"**

### Примерен вход и изход

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 95 | Less than 100 | 120 | Between 100 and 200 | 210 | Greater than 200 |

## Информация за скоростта

Да се напише функция, която **получава скорост** **(реално число)** и отпечатва **информация за скоростта**.

* При скорост **до 10** (включително) отпечатайте **"slow"**
* При скорост **над 10** и **до 50** (включително) отпечатайте **"average"**
* При скорост **над 50** и **до 150** (включително) отпечатайте **"fast"**
* При скорост **над 150** и **до 1000** (включително) отпечатайте **"ultra fast"**
* При по-висока скорост отпечатайте **"extremely fast"**

### Примерен вход и изход

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| 8 | slow | 49.5 | average | 126 | fast |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |
| 160 | ultra fast | 3500 | extremely fast |

## Лица на фигури

Да се напише функция, която **получава като вида и размерите на геометрична** фигура и пресмята лицето й. Фигурите са четири вида: квадрат (**square**), правоъгълник (**rectangle**), кръг (**circle**) и триъгълник (**triangle**). На първия ред на входа се чете вида на фигурата (текст със следните възможности: square, rectangle, circle или triangle).

* Ако фигурата е **квадрат (**square): на следващия ред се чете едно дробно число - дължина на страната му
* Ако фигурата е **правоъгълник (**rectangle): на следващите два реда четат две дробни числа - дължините на страните му
* Ако фигурата е **кръг (**circle): на следващия ред чете едно дробно число - радиусът на кръга
* Ако фигурата е **триъгълник (**triangle): на следващите два реда четат две дробни числа - дължината на страната му и дължината на височината към нея

Резултатът да се закръгли до **3 цифри след десетичната запетая**.

### Примерен вход и изход

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |  | **вход** | **изход** |
| "square",  5 | 25.000 | "rectangle",  7,  2.5 | 17.500 | "circle",  6 | 113.097 | "triangle"  4.5,  20 | 45.000 |