# Lab: Conditional Statements

Problems for in-class and homework exercises for the course ["Programming Basics" @ SoftUni](https://softuni.org/).

Test your solutions in the **Judge** system: <https://judge.softuni.org/Contests/Compete/Index/3489>

## Excellent Result

The first task in this chapter is to write a function isExcellent(grade) that **reads a grade** received as an argument and prints **"Excellent!"** if the grade is **5.50** or higher. If not – Print **“No”**

**Sample Input and Output**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Input** | **Output** |  | **Input** | **Output** |  | **Input** | **Output** |  | **Input** | **Output** |
| isExcellent(6) | Excellent! | isExcellent(5) | No | isExcellent(5.50) | Excellent! | isExcellent(5.49) | No |

### Hints and Guidelines

1. Create a **new JavaScript file** in the existing folder and name it appropriately. It is recommended to name each script file as the name of the task whose solution it contains.





1. The content of the new file will open in the window on the right.



1. Go to the file **isExcellent.js** and create the function **isExcellent(input):**

Картина, която съдържа текст

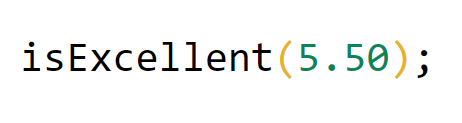
Описанието е генерирано автоматично

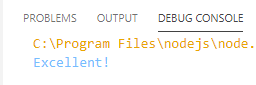
1. Check the value of the grade. If it is greater than or equal to 5.50, print the output according to the condition:

Text

Description automatically generated

1. Call the function with different input values and start it with **Ctrl + F5**:





## Greater Number

Write a function greaterNumber(firstNum,secondNum) that receives **two integers** and prints **the greater of the two**.

**Sample Input and Output**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Input** | **Output** |  | **Input** | **Output** |  | **Input** | **Output** |  | **Input** | **Output** |
| greaterNumber(5,3) | 5 | greaterNumber(3,5) | 5 | greaterNumber(10,10) | 10 | greaterNumber(-5,5) | 5 |

### Hints and Guidelines

1. Compare whether the first number is greater than the second number. Print the greater number.

Картина, която съдържа текст

Описанието е генерирано автоматично

## Even or Odd

Write a function oddOrEven(num) that receives **an integer** as an argument and prints to the console, whether it is **even** or **odd**. If it is **even**, print **"even"**. If it is **odd**, print "**odd"**.

**Sample Input and Output**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Input** | **Output** |  | **Input** | **Output** |  | **Input** | **Output** | |  | **Input** | **Output** |
| oddOrEven(2) | even | oddOrEven(3) | odd | oddOrEven(25) | | odd | oddOrEven(1024) | even |

### Hints and Guidelines

1. First, add a **new JavaScript** file to the existing project.
2. Check if the number is even by dividing it by 2 and checking if there is a remainder from the division. Print the output according to the condition - the text "**even**" or "**odd**".

Картина, която съдържа текст

Описанието е генерирано автоматично

## Password Guess

Write a function login(pass) that **receives a password** (text) and checks if the given password **matches** the phrase "**s3cr3t!P@ssw0rd**". If matches, output "**Welcome**". If it does not match, output "**Wrong password!**".

**Sample Input and Output**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Input** | **Output** |  | **Input** | **Output** |  | **Input** | **Output** |
| login("qwerty") | Wrong password! | login("s3cr3t!P@ssw0rd") | Welcome | login("s3cr3t!p@ss") | Wrong password! |

## Number 100...200

Write a function solve(num) that **receives an integer** and checks whether it is **less than 100**, **between 100 and 200**, or **greater than 200**. If the number is:

* less than 100 print: **"Less than 100"**
* between 100 and 200 print: **"Between 100 and 200"**
* greater than 200 print: **"Greater than 200"**

**Sample Input and Output**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Input** | **Output** |  | **Input** | **Output** |  | **Input** | **Output** |
| solve(95) | Less than 100 | solve(120) | Between 100 and 200 | solve(210) | Greater than 200 |

## Speed Info

Write a function solve(speed) that **receives speed (a floating-point number)** and prints **info about the speed.**

* At speed **up to 10** (inclusive) print **"slow"**
* At speed **greater than 10** and **up to 50** (inclusive) print **"average"**
* At speed **greater than 50** and **up to 150** (inclusive) print **"fast"**
* At speed **greater than 150** and **up to 1000** (inclusive) print **"ultra fast"**
* At **higher** speed print **"extremely fast"**

**Sample Input and Output**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Input** | **Output** |  | **Input** | **Output** |  | **Input** | **Output** |
| solve(8) | slow | solve(49.5) | average | solve(126) | fast |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Input** | **Output** |  | **Input** | **Output** |
| solve(160) | ultra fast | solve(3500) | extremely fast |

## Area of Figures

Write a function calculate(shape, …) that **receives the type and the dimensions of a geometric figure** and calculates its area. The figures are four types: **square**, **rectangle**, **circle**, and **triangle**. The first parameter of the input represents the type of the figure (a string with the following options: **square**, **rectangle**, **circle**, and **triangle**) and the next parameters are dependent on the type of figure.

* If the figure is a square: the second parameter is **a fractional number** – the length of the square side
* If the figure is a rectangle: the next two parameters **are both fractional numbers** – the lengths of its sides
* If the figure is a circle: the next parameter is **a fractional number** – the radius of the circle
* If the figure is a triangle: the next two parameters **are both fractional numbers** – the length of its side and the length of the height to it

Round the result to **3 digits after the decimal point**.

**Sample Input and Output**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Input** | **Output** |  | **Input** | **Output** |  | **Input** | **Output** |  | **Input** | **Output** |
| calculate(square, 5) | 25.000 | calculate(rectangle, 7, 2.5) | 17.500 | calculate(circle, 6) | 113.097 | calculate(triangle, 4.5, 20) | 45.000 |