

Atelier : Instructions conditionnelles

Tâches pour l'exercice en classe et pour les devoirs au cours « Bases de la [programmation](#) » @ SoftUni.

Тествайте решенията си в Judge системата : <https://judge.softuni.org/Contests/2401/Conditional-Statements-Lab>

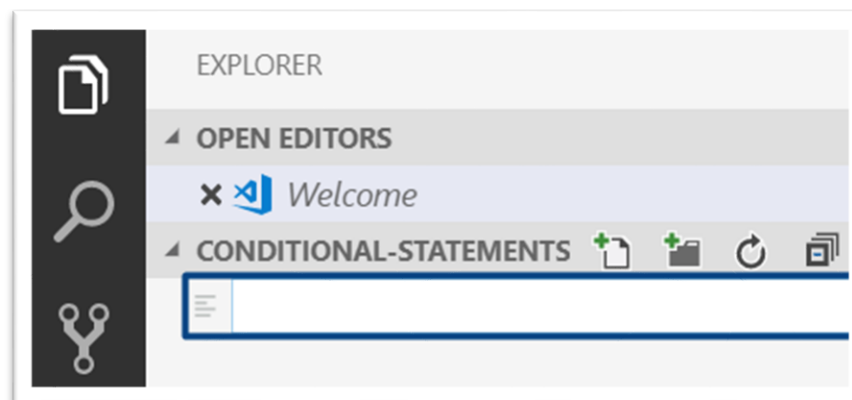
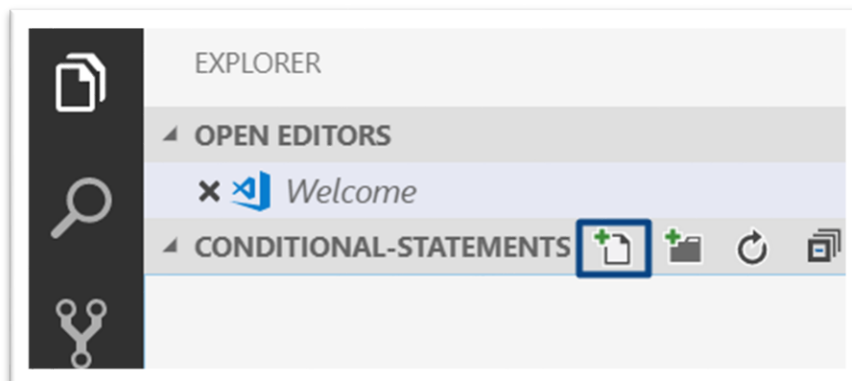
1. Excellente note

La première tâche de cette rubrique consiste à écrire **une fonction** qui **lit un score** reçu en argument et affiche « **Excellent !** » si le score est **supérieur ou égal** à 5,50.

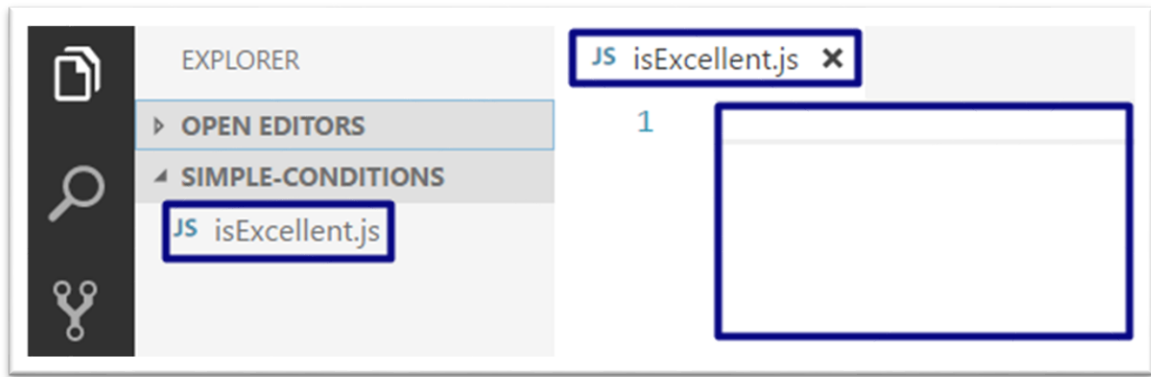
entrée	sortie	entrée	sortie	entrée	sortie	entrée	sortie
(["6"])	Excellente !	(["5"])	(pas de sortie)	(["5.50"])	Excellente !	(["5.49"])	(pas de sortie)

Lignes directrices

1. Créez **un nouveau fichier JavaScript** dans le dossier existant et nommez-le de manière appropriée. Il est recommandé que chaque fichier de script soit indiqué comme le nom de la tâche dont il contient la solution.



2. Le contenu du nouveau fichier s'ouvrira dans la fenêtre de droite.



3. Accédez au fichier **isExcellent.js** et créez la fonction **isExcellent(input)** :

```
function isExcellent(input) {  
  
}
```

4. Vérifiez la valeur de l'estimation. S'il est supérieur ou égal à 5,50, imprimez la sortie en fonction de la condition :

```
function isExcellent(input) {  
    let grade = Number(input[0]);  
    if (grade >= 5.50) {  
        console.log("Excellent!");  
    }  
}
```

5. Appelez la fonction avec des valeurs d'entrée différentes et exécutez-la avec **Ctrl + F5** :

```
isExcellent(["5.50"]);
```

PROBLEMS OUTPUT DEBUG CONSOLE

```
C:\Program Files\nodejs\node.  
Excellent!
```

```
isExcellent(["5.49"]);
```

PROBLEMS OUTPUT DEBUG CONSOLE TEST

C:\Program Files\nodejs\node.exe

2. Le plus grand nombre

Écrivez une fonction qui reçoit **deux entiers** et affiche **le plus grand des deux**.

Entrée et sortie d'échantillons

entrée	sortie
(["5", "3"])	5

entrée	sortie
(["3", "5"])	5

entrée	sortie
(["10", "10"])	10

entrée	sortie
(["-5", "5"])	5

Lignes directrices

1. Comparez si le premier nombre **num1** est supérieur au deuxième **num2**. Imprimez le nombre le plus élevé.

```
function greaterNumber(input) {  
  let num1 = Number(input[0]);  
  let num2 = Number(input[1]);  
  if (num1 > num2) {  
    console.log(num1);  
  } else {  
    console.log(num2);  
  }  
}
```

3. Pair ou impair

Écrivez une fonction qui reçoit **un entier** en argument et qui s'affiche sur la console, qu'elle soit paire ou **impaire**. S'il est **pair**, affiche « **pair** » s'il est impair « **impair** ».

Entrée et sortie d'échantillons

entrée	sortie
--------	--------

entrée	sortie
--------	--------

entrée	sortie
--------	--------

entrée	sortie
--------	--------

(["2"])	même	(["3"])	odd	(["25"])	odd	(["1024"])	même
---------	------	---------	-----	----------	-----	------------	------

Lignes directrices

1. Tout d'abord, ajoutez **un nouveau fichier JavaScript** au projet existant
2. Vérifiez si le nombre est pair en le divisant par 2 et vérifiez s'il reste de la division. Imprimez la sortie par condition – le texte « **pair** » ou « **impair** ».

```
function oddOrEven(input) {
  let num = Number(input[0]);
  if (num % 2 == 0) {
    console.log("even");
  } else {
    console.log("odd");
  }
}
```

4. Devinez le mot de passe

Écrivez une fonction qui **reçoit un mot de passe** (texte) et vérifie que le mot de passe donné **correspond** à la phrase "s3cr3t ! P@ssw0rd". En cas de coïncidence, affichez « **Bienvenue** ». En cas de non-coïncidence, affichez le message « **Mot de passe erroné !** ».

Entrée et sortie d'échantillons

entrée	sortie	entrée	sortie	entrée	sortie
(["qwerty"])	Mot de passe erroné !	(["S3Cr3T ! P@ssw0rd"])	Bienven ue	(["s3cr3t !p@ss"])	Mot de passe erroné !

5. Nombre de 100 à 200

Écrivez une fonction qui **reçoit un entier** et vérifie s'il est **inférieur à 100**, **compris entre 100 et 200** ou **supérieur à 200**. Si le nombre est :

- Moins de 100 caractères : « **Moins de 100** »
- Entre 100 et 200 tapez : « **Entre 100 et 200** »
- plus de 200 caractères : « **Plus de 200** »

Entrée et sortie d'échantillons

entrée	sortie
(["95"])	Moins de 100

entrée	sortie
(["120"])	Entre 100 et 200

entrée	sortie
(["210"])	Plus de 200

6. Informations sur la vitesse

Écrivez une fonction qui **obtient une vitesse (nombre réel)** et affiche les **informations de vitesse**.

- À des vitesses allant jusqu'à 10 (inclus), impression « **lente** »
- À des vitesses supérieures à 10 et jusqu'à 50 (inclus), imprimez « **moyenne** »
- À des vitesses supérieures à 50 et jusqu'à 150 (inclus), imprimez « **rapidement** »
- À des vitesses supérieures à 150 et jusqu'à 1000 (inclus), imprimez « **ultra rapide** »
- À des vitesses plus élevées, imprimez « **extrêmement vite** »

Entrée et sortie d'échantillons

entrée	sortie
(["8"])	lent

entrée	sortie
(["49.5"])	moyenne

entrée	sortie
(["126"])	rapide

entrée	sortie
(["160"])	ultra rapide

entrée	sortie
(["3500"])	Extrêmement rapide

7. Visages de personnages

Écrivez une fonction qui **reçoit comme le type et les dimensions d'une figure géométrique** et calcule sa face. Les figures sont de quatre types : **carré**, **rectangle**, **cercle** et **triangle**. La première ligne de l'entrée lit le type de la figure (texte avec les possibilités suivantes : **carré**, **rectangle**, **cercle** ou **triangle**).

- Si la figure est **carrée** : la ligne suivante indique un nombre fractionnaire - longueur de son côté
- S'il s'agit d' **un rectangle** : les deux lignes suivantes indiquent deux nombres fractionnaires, c'est-à-dire la longueur de ses côtés
- S'il s'agit d'**un cercle (cercle)** : sur la ligne suivante, on lit un nombre fractionnaire - le rayon du cercle
- S'il s'agit d'**un triangle** : les deux rangées suivantes indiquent deux nombres fractionnaires - la longueur de son côté et la longueur de la hauteur jusqu'à lui

Arrondissez le résultat à **3 décimales**.

Entrée et sortie d'échantillons

entrée	sortie
(["carré », « 5"])	25.000

entrée	sortie
(["rectangle » , "7", "2.5"])	17.500

entrée	sortie
(["cercle » , "6"])	113.097

entrée	sortie
(["triangle », "4.5", "20"])	45.000

