

Formation au Langage Kotlin

Sommaire

1. Présentation de Kotlin	3
2. Les littéraux	5
3. Base de la programmation en Kotlin	5
4. Les variables	6
5. Les constantes	9
6. Les commentaires	9
7. Conventions de codage	10
8. Nommer les variables	10
9. Sortie standard	11
10. Les types	12
10.1. Types des données	12
10.2. Classifications des types basiques	13
10.3. Conversions de types	14
10.4. Type d'une expression numérique	16
11. Les strings	16
11.1. Les bases du type String	16
11.2. Les templates	17
11.3. Travailler avec les Strings	19
12. Entrée standard avec <code>java Scanner</code>	22
13. Lire les données avec <code>readln()</code>	22
14. Les opérations	22
14.1. Opérations logiques	23
14.2. Opérations arithmétiques	24
14.3. Incrémentation & décrémentation	25
14.4. Les entiers en action	25
14.5. Les flottants	28
14.6. Les opérateurs de comparaison	29
14.7. Les caractères	31
15. Structures de contrôle	33
15.1. Structures conditionnelles avec <code>if</code>	33
15.2. Structures conditionnelles avec <code>when</code>	35
15.3. Structures itératives avec <code>repeat()</code>	37
15.4. Structures itératives avec <code>while</code> et <code>do ... while</code>	38
15.5. Structures itératives avec la boucle <code>for</code>	40
15.6. Les sauts & retours : <code>break</code> , <code>continue</code> , <code>return</code>	41
16. Listes mutables	43

16.1. Introduction	43
16.2. Travailler avec les listes mutables	44
16.3. Les listes à N dimensions	46
16.4. Boucles for sur les listes	49
17. Programmation procédurale	51
17.1. Appel d'une fonction	51
17.2. Déclaration d'une fonction	52
17.3. Arguments par défaut	55
17.4. Arguments nommés	58
17.5. Portées des variables	59
18. Les collections	61
18.1. Introduction aux collections	61
18.2. List (listes immuables)	61
18.3. Mutables List (listes mutables)	63
18.4. Maps (tableaux associatifs immuables)	66
18.5. Mutable Map (tableaux associatifs mutables)	69
18.6. Set (Sets immuables)	71
18.7. Mutable Set (Sets mutables)	72
19. La Programmation Orientée Objet (P.O.O.)	74
19.1. Déclaration d'une classe	74
19.2. Constructeurs (par défaut et primaire)	75
19.3. Fonctions membres (méthodes)	76
19.4. Accesseurs & Mutateurs	77
19.5. Constructeurs secondaires	78
20. Les tableaux	79
20.1. Le type Array	79
20.2. Les tableaux de Strings	80
20.3. Les tableaux à N dimensions	81
20.4. Boucle for sur les tableaux	82

Avant propos

Ce document présente le parcours de formation qui s'articule sur 3 types d'activités :

- **Introduction au concept :**

- Vous serez invité à lire via un lien une explication des concepts et savoirs abordés.
L'ensemble des sources proviennent du site de formation **JetBrains Academy** (<https://hyperskill.org/tracks>)

- **Réalisation de QCM :**

- Vous serez invité à vous réaliser pour chaque partie un ou plusieurs QCM via l'ENT.

- **Application au codage :**

- Vous serez invité à réaliser ou corriger des petits programmes Kotlin en utilisant l'**IDE IntelliJ**.

1. Présentation de Kotlin

- **Exposé des concepts & savoirs :**

- Lire la présentation officielle du langage **Kotlin** du site de la **JetBrains Academy** :
<https://hyperskill.org/learn/step/4350>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'ENT/**Pronote** : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 01-a : Hello Kotlin

Pour se mettre en forme nous vous demandons juste d'ouvrir dans **IntelliJ** le dossier-projet suivant : [./Learn_Kotlin_LdV/01-EXERCICES/01-Presentation_Kotlin/P01_a-Hello_Kotlin](#)

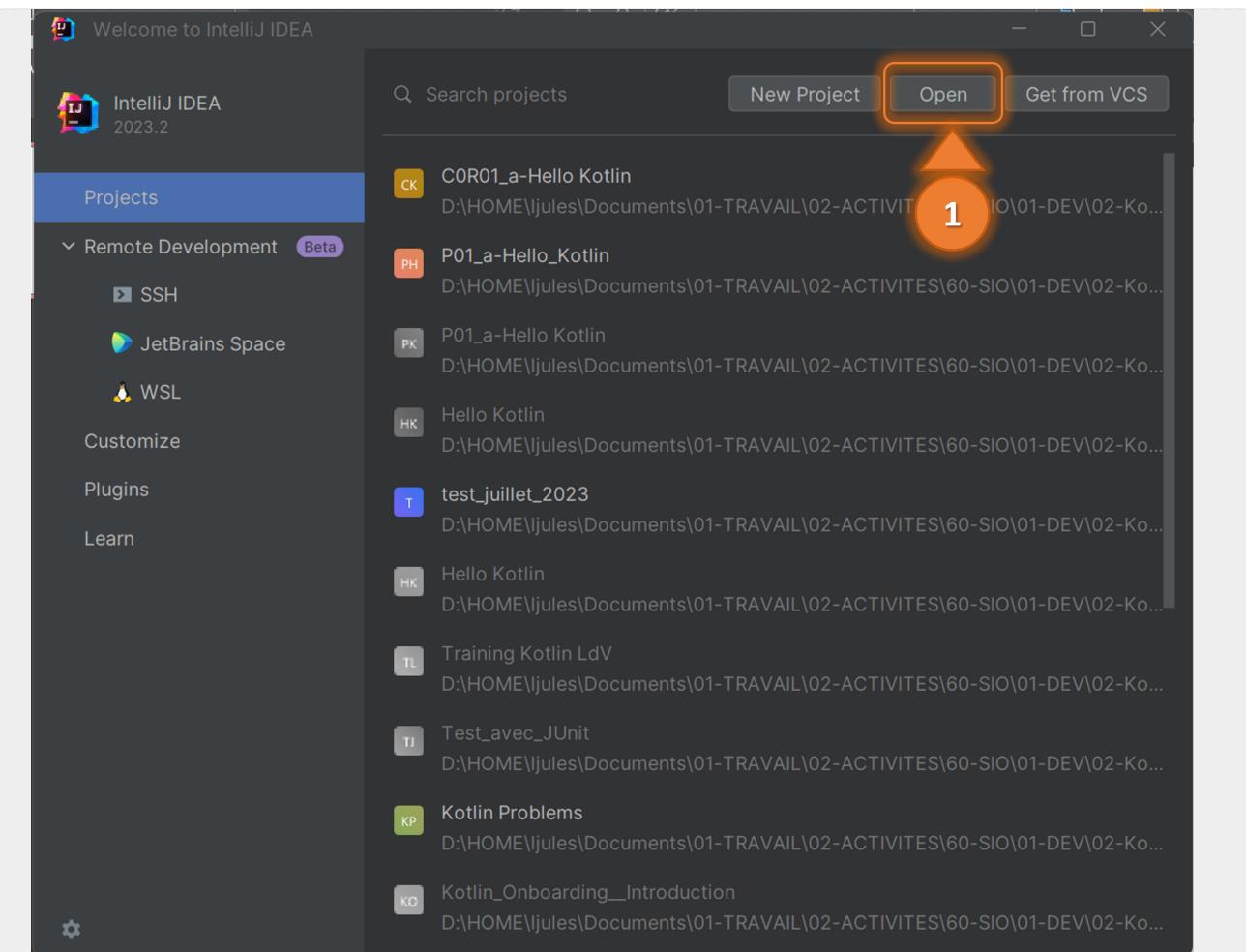
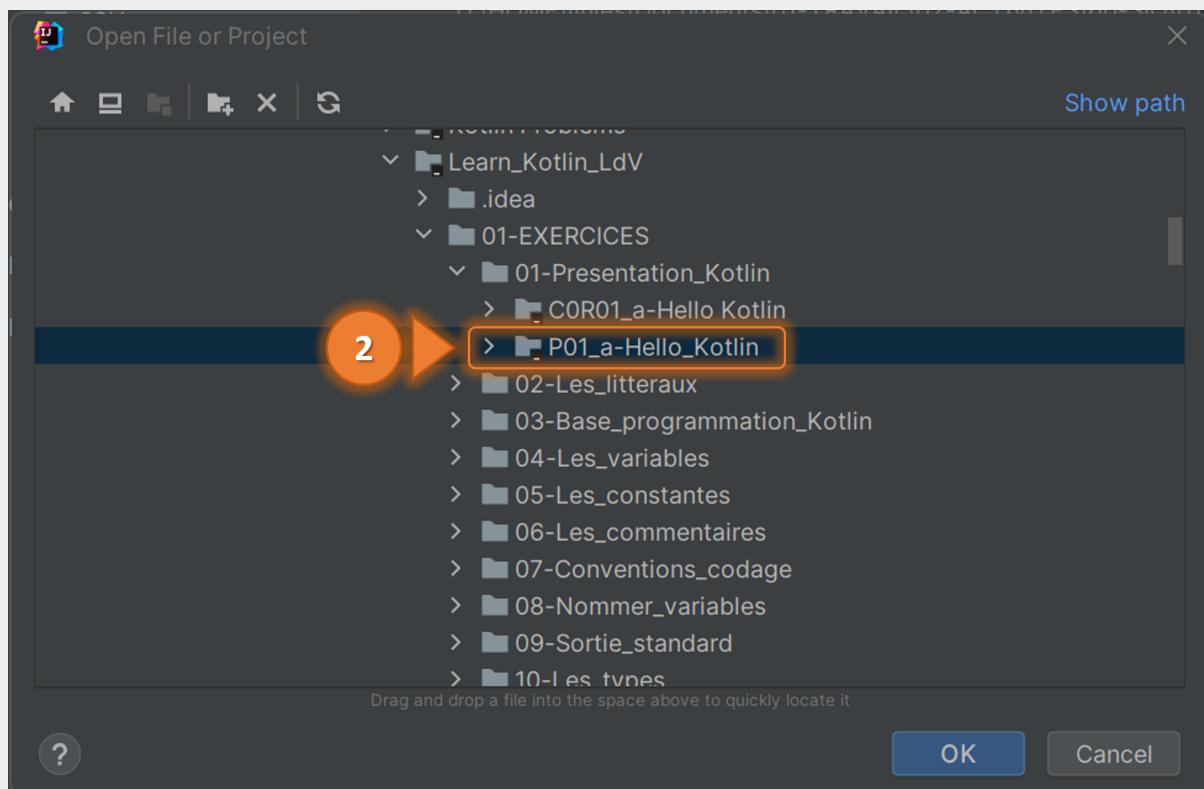
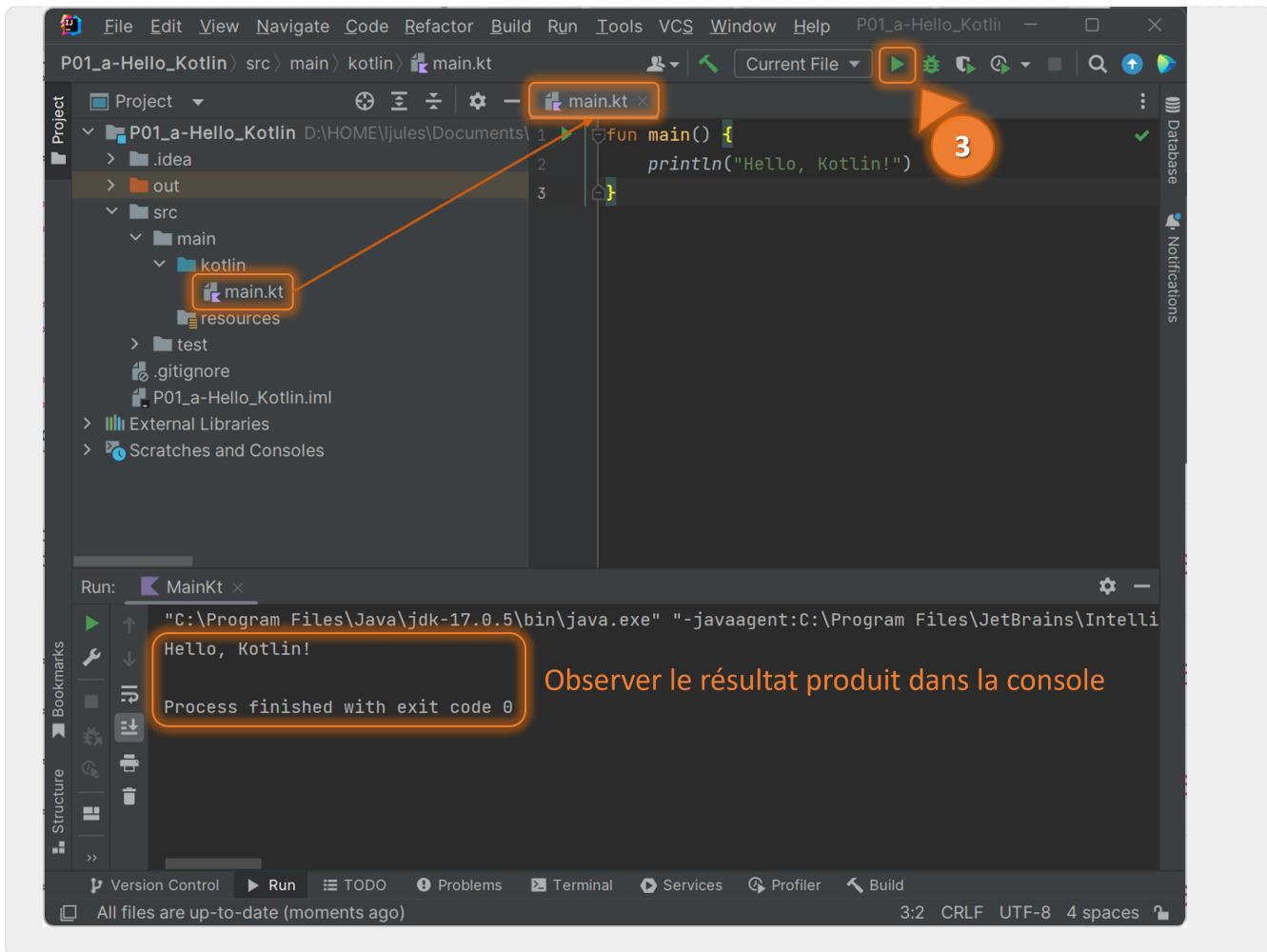


Figure 1. Marche à suivre pour exécuter le fichier





2. Les littéraux

- **Exposé des concepts & savoirs :**
 - Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/4351>
- **Validation de la compréhension par QCM :**
 - Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER
- **Mise en application :**
 - Il n'y a pas d'application à réaliser pour cette partie.

3. Base de la programmation en Kotlin

- **Exposé des concepts & savoirs :**
 - Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/4362>
- **Validation de la compréhension par QCM :**
 - Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER
- **Mise en application :**

Problème 03-a : Vérité éternelle

Enoncé :

Write a program that prints the string "2 + 2 = 4" (without quotes).

Sample Output

```
2 + 2 = 4
```

Fichier de travail :

```
./Learn_Kotlin_LdV/01-EXERCICES/01-Presentation_Kotlin/P03_a-Verite_eternel
```

Problème 03-b : I like Kotlin

Enoncé :

Write a program that prints the string "I like Kotlin" (without quotes).

Sample Output

```
I like Kotlin
```

Fichier de travail :

```
./Learn_Kotlin_LdV/01-EXERCICES/01-Presentation_Kotlin/P03_b-I_like_Kotlin
```

Problème 03-c : Plusieurs instructions

Enoncé :

Change the program so that it prints the words of the phrase "Learn Kotlin" on separate lines (make a program with multiple statements).

Sample Output

```
Learn  
Kotlin
```

Fichier de travail :

```
./Learn_Kotlin_LdV/01-EXERCICES/01-Presentation_Kotlin/P03_c-Plusieurs_instructions
```

4. Les variables

- Exposé des concepts & savoirs :

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/4371>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 04-a : Corriger le bug

Enoncé :

Given two variables **a** and **b** with invalid declarations. Fix these declarations. Ignore code style error for now.

Sample Output

```
512343 3431231
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\04-Les_variables\P04_a-Corriger_le_bug
```

Problème 04-b : Très grand nombre

Enoncé :

Declare a variable named **bigNumber** and initialize it with the integer value **100_000_000** (underscores are admissible here). Don't print anything else!

Sample Output

```
100000000
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\04-Les_variables\P04_b-Tres_grand_nombre
```

Problème 04-c : Corriger le programme

Enoncé :

You are given a program that must print the value of **string**, but it does not work. Please fix it.

Sample Output

```
Hello, Kotlin!
```

Fichier de travail :

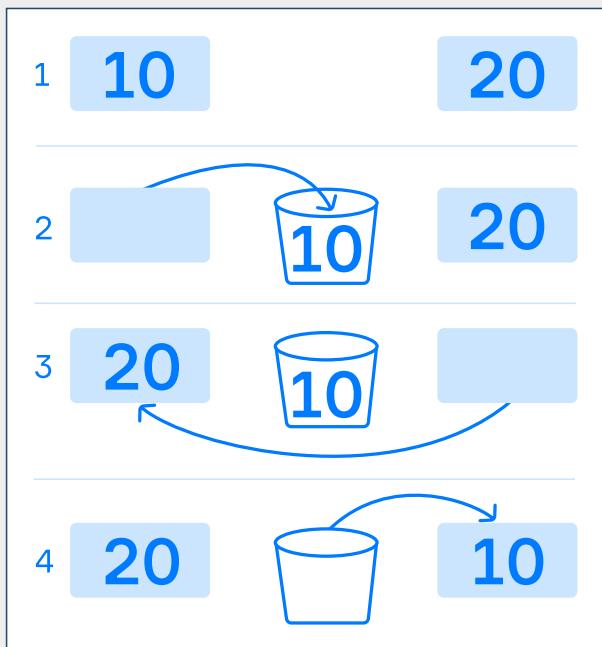
```
.\Learn_Kotlin_LdV\01-EXERCICES\04-Les_variables\P04_c-Corriger_programme
```

Problème 04-d : Echange

Enoncé :

Suppose you have two variables `a` and `b` that store some numbers. Swap the values of these variables. For example, if `a = 3` and `b = 5`, then after executing the code, `a` will contain `5`, and `b` will contain `3`.

Try to do it with the help of the third variable `c`. You can imagine that variables are just boxes with values, and you need to swap their contents, like this:



Sample Output

```
3 5  
5 3
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\04-Les_variables\P04_d-Echange
```

Problème 04-e : var ou val

Enoncé :

Here is a variable called `number`. Change its value to 5.

NOTE : There are at least two ways to do it.

Sample Output

```
5
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\04-Les_variables\\P04_e-var_ou_val

5. Les constantes

- Exposé des concepts & savoirs :

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/17408>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

Problème 05-a : Corriger l'erreur dans le code

Enoncé :

We wrote a program, but it doesn't work.

Please correct the error in the code.

The program must print [10000, 1000000, 100000000].

Sample Output

[10000, 1000000, 100000000]

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\05-Les_constants\\P05_a-Corriger_erreur_code

6. Les commentaires

- Exposé des concepts & savoirs :

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/4406>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

Problème 06-a : Commenter quelques lignes

Enoncé :

Here is a program that prints numbers. Comment some lines so that the program prints only these three numbers:

Résultat attendu :

5
6
7

Fichier de travail :

.\Learn_Kotlin_LdV\01-EXERCICES\06-Les_commentaires\P06_a-Commenter_quelques_lignes

Problème 06-b : Commentaires oubliés

Enoncé :

A programmer wrote a program with lots of comments in it but forgot to add the symbols required for comments. As a result, the program does not work. You need to comment some lines in this code to compile it.

Please do not remove the existing comments!

Résultat attendu :

It works!

Fichier de travail :

.\Learn_Kotlin_LdV\01-EXERCICES\06-Les_commentaires\P06_b-Commentaires_oubliés

7. Conventions de codage

- Exposé des concepts & savoirs :

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/4419>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

- Il n'y a pas d'application à réaliser pour cette partie.

8. Nommer les variables

- Exposé des concepts & savoirs :

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/4389>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

- Il n'y a pas d'application à réaliser pour cette partie.

9. Sortie standard

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/4425>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

- Il n'y a pas d'application à réaliser pour cette partie.

Problème 09-a : Besoin d'apprendre Kotlin

Enoncé :

Write a program that prints this text :

Résultat attendu :

```
1 | WE NEED
2 |
3 | TO LEARN KOTLIN
4 |
5 | AS QUICKLY AS POSSIBLE
```

Fichier de travail :

`.\Learn_Kotlin_LdV\01-EXERCICES\09-Sortie_standard\P09_a-Besoin_apprendre_Kotlin`

Problème 09-b : Corriger le code

Enoncé :

Take a look at the following Kotlin code snippet :

Code de départ :

```
fun main() {
    println(1, 2, 3...)
    println("Line print here ")
    println(42)
    println("Pay attention to ")
    println("syntax")
}
```

Without deleting any lines, correct the code so that you get the following output :

Résultat attendu en sortie :

```
1, 2, 3...
Line print here 42
Pay attention to syntax
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\09-Sortie_standard\\P09_b-Corriger_code

10. Les types

10.1. Types des données

- Exposé des concepts & savoirs :
 - Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/4388>
- Validation de la compréhension par QCM :
 - Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER
- Mise en application :

Problème 10_1-a : Corriger le programme

Enoncé :

There is a program that prints the value of `number`, but somehow it does not work. Please, fix it.

Note that there are at least two ways to do it.

Sample Output

```
100
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\10-Les_types\\10_1-Types_des_donnees\\P10_1_a-Corriger_programme

Problème 10_1-b : Penser comme un compilateur

Enoncé :

Declare two variables of type `String`, name them `first` and `last`, and assign your first name and your last name to them.

Declare one more variable of type **Int** with the name **age** and assign your age to it.

Sample Output

```
My name is John DOE and I'm 21 years old
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\10-Les_types\10_1-Types_des_donnees\P10_1_b-
Penser_compilateur
```

Problème 10_1-c : Editer-le

Enoncé :

The following code snippet contains errors. Find and correct them so that the final result will be the output of the following line :

"Kotlin is a cross-platform, statically typed, general-purpose programming language with type coercion."

Sample Output

```
Kotlin is a cross-platform, statically typed, general-purpose programming
language with type coercion.
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\10-Les_types\10_1-Types_des_donnees\P10_1_c-Editer_le
```

10.2. Classifications des types basiques

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/4455>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 10_2-a : Valeurs maximales des types entiers

Enoncé :

Write a program that prints max values of four integer types: **Int**, **Long**, **Short**, and **Byte** in **ascending order**.

Output each value on a new line.

Note :

Votre programme ne devra pas contenir directement les valeurs en durs, mais utiliser une

propriété/attribu/champs pour chaque type (**Int**, **Long**, etc.)

Résultat attendu en sortie :

```
127  
32767  
2147483647  
9223372036854775807
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\10-Les_types\10_2-  
Classification_types_basiques\P10_2_a-Valeurs_max_entiers
```

10.3. Conversions de types

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/4672>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'**ENT/Pronote** : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 10_3-a : Détecteur de mensonge

Enoncé :

Let's write a lie detector program! The program should take a string as an input, convert it to a boolean value, and print it as the result.

Exemple 1 :

Entrée :

```
0
```

Sortie :

```
false
```

Exemple 2 :

Entrée :

```
tRUE
```

Sortie :

```
true
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\10-Les_types\10_3-Conversions_types\P10_3_a-
Detecteur_mensonge
```

Problème 10_3-b : Convertisseur simple

Enoncé :

Let's make a simple converter that converts values to the following three types : **Int**, **Double**, and **Boolean**.

Your program should read a **value** and print the result of its conversion to **Int**, **Double**, and **Boolean** types sequentially.

Exemple 1 :

Entrée :

```
0
```

Sortie :

```
0
0.0
false
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\10-Les_types\10_3-Conversions_types\P10_3_b-
Convertisseur_simple
```

Problème 10_3-c : Force supérieure de l'empire

Enoncé :

It is known that the Galactic Empire has more ships than the Rebel Alliance and the number of the Empire's ships is a multiple of the number of the rebels' ships.

Write a program that calculates how many times larger the Empire fleet is.

Input : two integers as **String** (each starting on a new line). **Output :** **Int** value.

Exemple 1 :

Entrée :

```
2000  
1000
```

Sortie :

```
2
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\10-Les_types\10_3-Conversions_types\P10_3_c-  
Force_sup_empire
```

10.4. Type d'une expression numérique

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/9506>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

- Il n'y a pas d'application à réaliser pour cette partie.

11. Les strings

11.1. Les bases du type String

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/12553>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 11_1-a : Reslover le problème

Enoncé :

An inexperienced programmer created name incorrectly. You need to fix it so that the program outputs **10 years ago we were in London**.

Résultat attendu en sortie :

```
10 years ago we were in London
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\11-Les_strings\11_1-Bases_type_string\P11_1-a-  
Resolver_probleme
```

11.2. Les templates

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/4547>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 11_2-a : Afficher 2 nombres sur la même ligne

Enoncé :

Write a program that reads two integer numbers and prints them on the same line. The numbers are separated by a space.

Please solve the problem using string templates.

Exemple 1 :

Entrée :

```
8  
11
```

Sortie :

```
8 11
```

Exemple 2 :

Entrée :

```
-5  
0
```

Sortie :

```
-5 0
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\11-Les_strings\11_2-Les_templates\P11_2_a-
Afficher_2_nombres
```

Problème 11_2-b : Expression arithmétique

Enoncé :

Write a program that reads two numbers and prints their sum. The examples below demonstrate what the result should look like.

Exemple 1 :

Entrée :

```
12
34
```

Sortie :

```
12 plus 34 equals 46
```

Exemple 2 :

Entrée :

```
1
234
```

Sortie :

```
1 plus 234 equals 235
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\11-Les_strings\11_2-Les_templates\P11_2_b-
Expression_arithmetique
```

Problème 11_2-c : String et sa moitié

Enoncé :

Write a program that reads the **String** representation of an arbitrary number and outputs

a line with this number and its **Int** representation divided by 2.

Look at the example to understand the output format.

Input :

```
13
```

Output :

```
The obtained value is 13 and its Int representation after division by 2 is 6
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\11-Les_strings\11_2-Les_templates\P11_2_c-
String_et_sa_moitie
```

11.3. Travailler avec les Strings

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/12561>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 11_3-a : Vérité et opérations

Enoncé :

Read three lines. You need to check whether the third line is equal to the sum of the first two lines.

If the first two lines put together form the third one, print **true**; otherwise **false**.

Exemple 1

Input :

```
sum
mer
summer
```

Output :

```
true
```

Exemple 2

Input :

```
tomato  
cucumber  
salad
```

Output :

```
false
```

Fichier de travail :

`.\\Learn_Kotlin_LdV\\01-EXERCICES\\11-Les_strings\\11_3-Travailler_avec_strings\\P11_3_a-Verite_et_operations`

Problème 11_3-b : Informations au sujet d'une personne

Enoncé :

Write a program that reads the first name, the last name and the age of a person and then prints the information as in the examples below.

To solve this problem, use string templates.

Exemple 1

Input :

```
John Smith 30
```

Output :

```
J. Smith, 30 years old
```

Fichier de travail :

`.\\Learn_Kotlin_LdV\\01-EXERCICES\\11-Les_strings\\11_3-Travailler_avec_strings\\P11_3_b-Informations_personne`

Problème 11_3-c : Dizaine d'un nombre

Enoncé :

Ecrire un programme qui lit un nombre (qui sera supérieur ou égale à 10) et qui affiche le chiffre des dizaines.

Utiliser les index pour résoudre le problème !

Exemple 1

Input :

```
234
```

Output :

```
3
```

Fichier de travail :

`.\Learn_Kotlin_LdV\01-EXERCICES\11-Les_strings\11_3-Travailler_avec_strings\P11_3_c-Dizaine_un_nombre`

Problème 11_3-d : N-ième symbole

Enoncé :

Write a program that reads one string and one number N and prints the N-th symbol of the input string. The examples below show how the result should look like.

Input : one **String** and one number of type **Int** in separate lines.

Output : single **String**. Use the following template :

Symbol # N of the string "**input string**" is '**character**'

Where N is the input number; **input string** is the string your read, and character is the N-th symbol of the **input string**.

Exemple 1

Input :

```
hello world
```

```
7
```

Output :

```
Symbol # 7 of the string "hello world" is 'w'
```

Fichier de travail :

`.\Learn_Kotlin_LdV\01-EXERCICES\11-Les_strings\11_3-Travailler_avec_strings\P11_3_d-Nieme_symbole`

12. Entrée standard avec `java Scanner`

- Exposé des concepts & savoirs :

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/4445>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

Problème 12-a : Quatre nombres

Enoncé :

Write a program that reads four integer numbers from one line and prints them each on a new line. Input numbers should be separated by one or more spaces.

Input :

```
101 102 103 104
```

Output :

```
101  
102  
103  
104
```

Fichier de travail :

`.\Learn_Kotlin_LdV\01-EXERCICES\12-Entree_std_avec_Java_Scanner\P12_a-Quatre_nombres`

13. Lire les données avec `readln()`

- Exposé des concepts & savoirs :

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/11224>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

- Pas d'application à réaliser pour cette partie.

14. Les opérations

14.1. Opérations logiques

- Exposé des concepts & savoirs :

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/4515>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

Problème 14_1-a : Invitations et cadeaux

Enoncé :

Mr. Groundhog is throwing a party. He's known for his pragmatism, so only those who bring a gift will be allowed to attend the party. And, of course, a guest must have an invitation.

Write a program that checks if a guest should be allowed at the party. The program should read **two booleans**, each on a **separate line** :

- One showing whether the guest has an invitation;
- Another indicating if the guest brought a gift.

Print **true** if both conditions are met and **false** if not

Exemple 1 :

Input :

```
true  
true
```

Output :

```
true
```

Exemple 2 :

Input :

```
true  
false
```

Output :

```
false
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\14-Les_operations\\14_1-Operations_logiques\\P14_a-
Invitations_et_cadeaux

Problème 14_1-b : Coder une expression booléenne

Enoncé :

Write a program that reads three boolean variables `x`, `y`, and `z` (each on a separate line) and then prints the result of the following logical expression: `NOT(x AND y) OR (z)`.

Exemple 1 :

Input :

```
true  
true  
false
```

Output :

```
false
```

Exemple 2 :

Input :

```
true  
false  
true
```

Output :

```
true
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\14-Les_operations\\14_1-Operations_logiques\\P14_1_b-
Coder_expression_bool

14.2. Opérations arithmétiques

- Exposé des concepts & savoirs :

- Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/4472>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

- Aucune application à résoudre pour cette partie.

14.3. Incrémentation & décrémentations

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/10776>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

- Aucune application à résoudre pour cette partie.

14.4. Les entiers en action

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/4488>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 14_4-a : Partager équitablement des noisettes

Enoncé :

N squirrels found K nuts and decided to split them equally. Calculate how many nuts go for each squirrel.

Input data format : There are two positive numbers N and K, each of them is not greater than 10 000.

Input

```
3  
14
```

Output

```
4
```

Fichier de travail :

[.\Learn_Kotlin_Ldv\01-EXERCICES\14-Les_operations\14_4-Entiers_en_action\P14_4_a-Partager_noisettes](.\\Learn_Kotlin_Ldv\\01-EXERCICES\\14-Les_operations\\14_4-Entiers_en_action\\P14_4_a-Partager_noisettes)

Problème 14_4-b : Noisettes restantes

Enoncé :

N squirrels found K nuts and decided to split them equally. Find how many nuts will be left after each of the squirrels takes an equal amount of nuts

Input data format : There are two positive numbers N and K, each of them is not greater than 10 000.

Input

```
3  
14
```

Output

```
2
```

Fichier de travail :

`.\\Learn_Kotlin_LdV\\01-EXERCICES\\14-Les_operations\\14_4-Entiers_en_action\\P14_4_b-Noisettes_restantes`

Problème 14_4-c : Nombre pair suivant

Enoncé :

Given a natural number N, not greater than 10 000. Output the even number that follows this N.

Tip : Try to write a short formula using the modulo operator %

Exemple 1 :

Input

```
7
```

Output

```
8
```

Exemple 2 :

Input

```
8
```

Output

10

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\14-Les_operations\\14_4-Entiers_en_action\\P14_4_c-Nombre_pair_suivant

Problème 14_4-d : Somme des chiffres

Enoncé :

There is a three-digit integer (an integer from 100 to 999). Find the sum of its digits.

Don't forget that you can get digits in a number with **/10** and **%10**.

Exemple :

Input

476

Output

17

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\14-Les_operations\\14_4-Entiers_en_action\\P14_4_d-Somme_des_chiffres

Problème 14_4-e : Formatage de dates

Enoncé :

You are given a number that represents the number of seconds passed since **1/1/1970**.

Write a program that calculates the current time, i.e., finds the hours, minutes, and seconds of the given number of seconds.

Format : hours:minutes:seconds

Example : 14:9:7

You don't have to import anything, just use % and /, and remember how long one day is.

You do not need to print the number of days and do not think about UTC and GMT.

Exemple 1 :

Output

15:47:4

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\14-Les_operations\\14_4-Entiers_en_action\\P14_4_e-Formatage_date

14.5. Les flottants

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/4502>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 14_5-a : Evaluer une expression

Enoncé :

Write a program that reads four double values **a**, **b**, **c**, **d** and calculates this expression :

- **a * 10.5 + b * 4.4 + (c + d) / 2.2**

Exemple 1 :

Input :

```
1  
2.5  
0  
4.4
```

Output :

23.5

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\14-Les_operations\\14_5-Les_flaotants\\P14_5_a-Evaluer_expression

Problème 14_5-b : Celsius vers Fahrenheit

Enoncé :

Write a program that reads a temperature in Celsius ($^{\circ}\text{C}$) and shows its equivalent in Fahrenheit temperature ($^{\circ}\text{F}$).

Use this formula :

```
* fahrenheit = celsius * 1.8 + 32
```

Exemple 1 :

Input :

```
32.9
```

Output :

```
91.22
```

Exemple 2 :

Input :

```
0
```

Output :

```
32.0
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\14-Les_operations\14_5-Les_flottants\P14_5-b-
Celsius_vers_Farenheit
```

14.6. Les opérateurs de comparaison

- Exposé des concepts & savoirs :

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/4525>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

Problème 14_6-a : Vérifier la valeur

Enoncé :

Write a program that reads a value and checks whether it is less than 10 but greater than 0.

Exemple :

Input :

0

Output :

false

Fichier de travail :

.\Learn_Kotlin_LdV\01-EXERCICES\14-Les_operations\14_6-Operateurs_comparaison\P14_6_a-Verifier_valeur

Problème 14_6-b : Tailles croissantes ou décroissantes

Enoncé :

Imagine there are three boys in a sports class.

You need to write a program that checks if the boys are arranged in ascending or descending order by height. The program should read three integer numbers **h1**, **h2**, **h3** and output **true** or **false**. If the boys have the same height, they are arranged correctly.

Exemple 1 :

Input :

165
161
158

Output :

true

Exemple 2 :

Input :

155

165
160

Output :

false

Exemple 3 :

Input :

161
161
165

Output :

true

Fichier de travail :

. \Learn_Kotlin_LdV\01-EXERCICES\14-Les_operations\14_6-Operateurs_comparaison\P14_6_b-Tailles_croissantes_decroissantes

14.7. Les caractères

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/4680>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'ENT/**Pronote** : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 14_7-a : Comparaison de caractères

Enoncé :

Write a program that reads two Latin letters as characters and compares them ignoring cases. If these characters represent the same letter print **true** , otherwise **false**.

Exemple 1 :

Input :

a
b

Output :

```
false
```

Exemple 1 :

Input :

```
d  
D
```

Output :

```
true
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\14-Les_operations\\14_7-Les_caracteres\\P14_7_a-
Comparaison_de_caracteres

Problème 14_7-b : Caractère précédent

Enoncé :

Write a program that reads four characters and prints the previous character in the Unicode table for each of them.

Exemple 1 :

Input :

```
b  
c  
d  
e
```

Output :

```
a  
b  
c  
d
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\14-Les_operations\\14_7-Les_caracteres\\P14_7_b-
Caractere_precedent

15. Structures de contrôle

15.1. Structures conditionnelles avec **if**

- Exposé des concepts & savoirs :

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/4625>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

Problème 15_1-a : Positif, négatif ou nul

Enoncé :

Write a program that reads an integer from the console and prints :

- if the number < 0 : negative;
- if the number > 0 : positive;
- if the number = 0 : zero;

Exemple 1 :

Input :

-5

Output :

negative

Fichier de travail :

`.\Learn_Kotlin_LdV\01-EXERCICES\15-Structures_de_controle\15_1-if\P15_1_a-
Positif_negatif_nul`

Problème 15_1-b : Sommeil et santé

Enoncé :

Ann watched a TV program about health and learned that she should sleep at least **A** hours per day, but oversleeping is also not healthy and you should not sleep more than **B** hours.

Now Ann sleeps **H** hours per day. If Ann's sleep schedule complies with the requirements of that TV program print "**Normal**". If Ann sleeps less than **A** hours, output "**Deficiency**", if she sleeps more than **B** hours, output "**Excess**".

Input to this program are the three strings with variables in the following order : **A**, **B**, **H**. **A** is always less than or equal to **B**.

Please note letters case: the output should exactly correspond to what is required in the problem, i.e. if the program has to output "Excess", such output as "excess", "EXCESS", "ExCeSS" and others will not be considered correct.

You should think **carefully** about all the conditions, which you need to use. A special attention should be paid to the strictness of the conditional operators used : distinguish between < and < =; > and > =.

In order to understand which ones to use, read carefully the problem statement.

Exemple 1 :

Input :

```
6  
10  
8
```

Output :

```
Normal
```

Exemple 2 :

Input :

```
7  
9  
10
```

Output :

```
Excess
```

Exemple 3 :

Input :

```
7  
9  
2
```

Output :

```
Deficiency
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\15-Structures_de_controle\\15_1-if\\P15_1_b-Sommeil_et_sante

Problème 15_1-c : Nombre de diviseurs

Enoncé :

Write a program that reads an integer number and checks if it is divisible by 2, 3, 5, or 6.

If the number is divisible by M, the program should output "**Divided by M**".

The program should check all the divisors listed above. The order of divisors in the result can be any.

Tip : Use the modulo operator `%` to check whether a number divided by another one.

Exemple 1 :

Input :

6

Output :

Divided by 2
Divided by 3
Divided by 6

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\15-Structures_de_controle\\15_1-if\\P15_1_c-nombre_diviseurs

15.2. Structures conditionnelles avec `when`

- Exposé des concepts & savoirs :

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/4631>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

Problème 15_2-a : Harry Potter

Enoncé :

Harry Potter needs our help to identify which traits characterize each house.

Read a string that represents a house and output the following :

- if it is **gryffindor**, output **bravery**;
- if it is **hufflepuff**, output **loyalty**;
- if it is **slytherin**, output **cunning**;
- if it is **ravenclaw**, output **intellect**;
- if otherwise, output **not a valid house**.

Exemple 1 :

Input :

gryffindor

Output :

bravery

Fichier de travail :

.\Learn_Kotlin_LdV\01-EXERCICES\15-Structures_de_controle\15_2-when\P15_2_a-Harry_Potter

Problème 15_2-b : Direction

Enoncé :

Write a program, that reads a direction number :

- 1 → "move up"
- 2 → "move down"
- 3 → "move left"
- 4 → "right"
- 0 → "do not move"

If the input does not correspond to any of the listed directions, output "error!"

Output text should match the sample ! It is true for the letter case and spaces.

Exemple 1 :

Input :

1

Output :

```
move up
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\15-Structures_de_controle\15_2-when\P15_2_b-Direction
```

15.3. Structures itératives avec `repeat()`

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/4643>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 15_3-a : Compter les nombres positifs

Enoncé :

Write a program that reads a sequence and prints how many positive numbers it contains.

The first number is the length of the sequence. Other numbers are the elements of this sequence.

Exemple 1 :

Input :

```
8  
2  
3  
0  
7  
4  
-2  
-3  
0
```

Output :

```
4
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\15-Structures_de_controle\15_3-  
Repetition_blocs\P15_3_a-Compter_nombres_positifs
```

Problème 15_3-b : Notes

Enoncé :

Find how many "D", "C", "B" and "A" grades the students have got after the last Computer Science test. There are **n** students in the class. The program gets the **n** number as input and then gets the grades one by one. The program should output four numbers in a single line, representing the grades from "D" to "A".

Numbers represent the following grades: 2 is "D", 3 is "C", 4 is "B", and 5 is "A".

Exemple 1 :

Input :

```
10  
3  
5  
4  
4  
2  
5  
5  
5  
2
```

Output :

```
2 1 2 5
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\15-Structures_de_controle\15_3-  
Repetition_blocs\P15_3_b-Notes
```

15.4. Structures itératives avec **while** et **do ... while**

- Exposé des concepts & savoirs :

- Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/4659>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

Problème 15_4-a : Calcul des intérêts

Enoncé :

When a bank has financial problems, the government can return a client's deposit if it is less than **700 000**. The interest rate for a particular deposit is **7,1 %** a year. Interest is added to the same deposit at the end of the year, and then a new value of the interest is calculated.

Find out **how many years** it will take for the sum of the deposit to exceed the value protected by the government.

The input format : The initial sum of the deposit.

It is guaranteed that the value will be between 50,000 and 700,000.

Please note that the deposit amount increases by a value proportional to the interest rate each year – in this case, 1,071 (7,1 %). Thus, the formula for calculating the deposit amount will look like this :

deposit = deposit * 1.071.

Exemple 1 :

Input :

650000

Output :

2

Fichier de travail :

`.\Learn_Kotlin_LdV\01-EXERCICES\15-Structures_de_controle\15_4-
while_ET_do_while\P15_4_a-Calcul_interets`

Problème 15_4-b : Somme de valeurs

Enoncé :

Find the sum of all elements in a sequence, ending with 0.

The number 0 itself is not included in the sequence and serves as a sign that the sequence ended.

Although, if you add 0, nothing bad will happen :)

Exemple 1 :

Input :

```
3  
6  
8  
0
```

Output :

```
17
```

Fichier de travail :

`.\Learn_Kotlin_LdV\01-EXERCICES\15-Structures_de_controle\15_4-
while_ET_do_while\P15_4_b-Somme_valeurs`

15.5. Structures itératives avec la boucle **for**

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/4652>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 15_5-a : Somme des entiers d'un intervalle

Enoncé :

Read two integers **a** and **b**.

Print the sum of all the integers from **a** to **b** inclusive.

It is guaranteed that $a < b$.

Exemple 1 :

Input :

```
1  
24
```

Output :

```
300
```

Fichier de travail :

`.\Learn_Kotlin_LdV\01-EXERCICES\15-Structures_de_controle\15_5-Boucle_for\P15_5_a-`

Problème 15_5-b : Valeur minimale de N nombres

Enoncé :

Write a program that finds the minimum value of N numbers.

The first line contains the number N.

The other lines contain N numbers.

Output an integer number which is the minimum of N numbers.

Exemple 1 :

Input :

```
5
5
1
4
2
3
```

Output :

```
1
```

Fichier de travail :

.\Learn_Kotlin_LdV\01-EXERCICES\15-Structures_de_controle\15_5-Boucle_for\P15_5_b-Minimal_de_N_nombres

15.6. Les sauts & retours : **break**, **continue**, **return**

- Exposé des concepts & savoirs :

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/6165>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

Problème 15_6-a : Alphabet et mot

Enoncé :

Write a program that reads a word and prints all the alphabet letters that **are not used** in this word.

Only lowercase letters are counted.

Exemple 1 :

Input :

```
hello
```

Output :

```
abcdefghijklmnopqrstuvwxyz
```

Exemple 2 :

Input :

```
kotlin
```

Output :

```
abcdefghijklmnopqrstuvwxyz
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\15-Structures_de_controle\\15_6-
Sauts_et_retours\\P15_6_a-Alphabet_et_mot

Problème 15_6-b : Premier chiffre

Enoncé :

Write a program that reads a combination of letters and numbers and prints the first digit in it. It is guaranteed that there is at least one digit in the combination.

Exemple 1 :

Input :

```
abcd37lnj
```

Output :

```
3
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\15-Structures_de_controle\\15_6-
Sauts_et_retours\\P15_6_b-Premier_chiffre

16. Listes mutables

16.1. Introduction

- Exposé des concepts & savoirs :

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/14879>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

Problème 16_1-a : Initialisation d'une liste d'entiers

Enoncé :

Create an `Int` mutable list named `numbers` with five elements : **12, 17, 8, 101, and 33**. After that, the code will output it.

Use the provided code template.

Output :

```
12, 17, 8, 101, 33
```

Fichier de travail :

```
.\Learn_Kotlin_Ldv\01-EXERCICES\16-Listes_mutables\16_1-Introduction\P16_1_a-
Initialisation_liste_entiers
```

Problème 16_1-b : Initialisation d'une liste

Enoncé :

Create a mutable list of integers named `numbers` with 100 elements. Its first element must be 1, the tenth must be 10, and the hundredth one must be 100. All other elements must be equal to 0.

Use the provided code template.

Output :

```
1, 0, 0, 0, 0, 0, 0, 0, 0, 10, 0, 0, ... 100
```

Fichier de travail :

```
.\Learn_Kotlin_Ldv\01-EXERCICES\16-Listes_mutables\16_1-Introduction\P16_1_b-Init_list
```

Problème 16_1-c : Echange

Enoncé :

Suppose you have a `numbers` list of the `Int` type. Swap the first and the last elements of the list. It is guaranteed that the size of the list is larger than 0.

Exemple 1 :

Input :

```
1 2 3 4 5
```

Output :

```
5 2 3 4 1
```

Fichier de travail :

`.\\Learn_Kotlin_LdV\\01-EXERCICES\\16-Listes_mutables\\16_1-Introduction\\P16_1_c-Echange`

16.2. Travailler avec les listes mutables

- Exposé des concepts & savoirs :

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/14902>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

Problème 16_2-a : Encore une autre liste

Enoncé :

Create a string mutable list capitals with five elements: "Tokyo", "Moscow", "Paris", "Washington", "Beijing".

Output :

```
Tokyo, Moscow, Paris, Washington, Beijing
```

Fichier de travail :

`.\\Learn_Kotlin_LdV\\01-EXERCICES\\16-Listes_mutables\\16_2-Travailler_avec_listes_mutables\\P16_2_a-Encore_une_autre_liste`

Problème 16_2-b : Addition de listes

Enoncé :

Concatenate two mutable lists `firstList` and `secondList` and print the result.

In the example below, each line corresponds to a separate list. Elements are separated by spaces.

Tip : Use the function `joinToString()`.

Exemple 1 :

Input :

```
valar morghulis  
valar dohaeris
```

Output :

```
valar, morghulis, valar, dohaeris
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\16-Listes_mutables\16_2-  
Travailler_avec_listes_mutables\P16_2_b-Addition_de_listes
```

Problème 16_2-c : Garde au portail

Enoncé :

You need to help the guard who is watching the main gate leading to the wilderness beyond the Wall. It is his responsibility to put everyone who gets through the gate on the checklist `backToTheWall`.

Add the name of the watchman (stored in the `returnedWatchman` variable) to the list `backToTheWall` and print the result with `joinToString()`.

Exemple :

Input :

```
Benjen Stark, Samwell Tarly, Gared Tuttle  
Jon Snow
```

Output :

```
Benjen Stark, Samwell Tarly, Gared Tuttle, Jon Snow
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\16-Listes_mutables\\16_2-Travailler_avec_listes_mutables\\P16_2_c-Garde_au_portail

Problème 16_2-d : Jouer avec les nombres

Enoncé :

You have a mutable list of integers `numbers`. Add the sum of all list elements to the beginning of the list. Then delete the penultimate item in the list. It is guaranteed that the length of the array is greater than 2.

Exemple :

Input :

```
8 11 1 2 3
```

Output :

```
25 8 11 1 3
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\16-Listes_mutables\\16_2-Travailler_avec_listes_mutables\\P16_2_d-Jouer_avec_nombres

16.3. Les listes à N dimensions

- Exposé des concepts & savoirs :

- Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/15127>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

Problème 16_3-a : Listes Kotlin

Enoncé :

Write code that prints the third row of `inputList` in one string. The elements should be separated by a comma and space.

It is guaranteed that `inputList` is already initialized and contains at least 3 nested lists.

Tip : Use the `joinToString()` function.

Exemple 1 :

Input :

```
4
(-_-) Program
(_^_)/ with
(>^_^)> Kotlin!
wrong row
```

Output :

```
(>^_^)>, Kotlin!
```

Fichier de travail :

.\Learn_Kotlin_LdV\01-EXERCICES\16-Listes_mutables\16_3-Listes_N_dimensions\P16_3_a-
Listes_Kotlin

Problème 16_3-b : Renversement

Enoncé :

Imaginer que vous avez une liste 2D `inputList`. Il est garanti quelle comporte au moins 2 listes imbriquées.

Créer une nouvelle liste 2D `resultList`comportant 2 sous-listes. La première liste imbriquée (sous-liste) est la dernière liste imbriquée de 'inputList et la seconde liste imbriquée de resultList est la première de inputList.`

Afficher la liste résultat `resultList`.

Tip : Vous pouvez utiliser les méthodes `first()` and `last()`.

Exemple 1 :

Input :

```
3
00 01 02 03
10 11 12 13
20 21 22 23
```

Output :

```
[[20, 21, 22, 23], [00, 01, 02, 03]]
```

Fichier de travail :

.\Learn_Kotlin_LdV\01-EXERCICES\16-Listes_mutables\16_3-Listes_N_dimensions\P16_3_b-

Problème 16_3-c : Cube de zéros

Enoncé :

Create a 3x3x3 three-dimensional list that contains elements of type **Int** and is filled with zeros (0).

Print the list of lists in one string (just use **print()** or **println()** function).

Sortie attendue :

```
[[[0, 0, 0], [0, 0, 0], [0, 0, 0]], [[0, 0, 0], [0, 0, 0], [0, 0, 0]], [[0, 0, 0], [0, 0, 0], [0, 0, 0]]]
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\16-Listes_mutables\16_3-Listes_N_dimensions\P16_3_c-Cube_de_zeros
```

Problème 16_3-d : Coins

Enoncé :

Imagine you have a 2D list **inputList**, whose dimension is greater than or equal to two. Print all of its corner elements in the following order : left to right and top to bottom.

Print the result for two elements in one line. Use a single space to separate two elements on the same line.

Take a look at the examples below and remember that nested lists can be of different lengths !

Tip : You can use **first()** and **last()**.

Exemple 1 :

Input :

```
3
1 0 1
0 0 0
1 0 1
```

Output :

```
1 1
1 1
```

Exemple 2 :

Input :

```
2  
2 2  
2 2
```

Output :

```
2 2  
2 2
```

Fichier de travail :

.\Learn_Kotlin_LdV\01-EXERCICES\16-Listes_mutables\16_3-Listes_N_dimensions\P16_3_d-Coins

16.4. Boucles **for** sur les listes

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/15022>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 16_4-a : Nombre dans une séquence

Enoncé :

Write a program that checks if a set of N numbers contains a number M.

The first line contains the N number.

The next N lines contain the set of numbers one number per line.

The last line contains one integer number M.

You need to output either YES or NO.

Exemple 1 :

Input :

```
3  
1  
3  
2
```

3

Output :

YES

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\16-Listes_mutables\\16_4-
Boucles_for_sur_listes\\P16_4_a-Nombre_dans_une_sequence

Problème 16_4-b : Nombre d'occurrences

Enoncé :

Write a program that counts the times a number **M** occurs in **N** numbers.

- The first line contains the **N** number.
- The next **N** lines contain the numbers.
- The last line contains the **M** number.

Output only a single non-negative integer number.

Exemple 1 :

Input :

6
1
2
3
4
2
1
2

Output :

2

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\16-Listes_mutables\\16_4-
Boucles_for_sur_listes\\P16_4_b-Nombre_occurrences

Problème 16_4-c : Triples

Enoncé :

Write a program that reads a list of integers and outputs the number of triples in the list.

A triple is three consecutive integers in ascending order. 3, 4, 5 is a triple, but 5, 4, 3 and 2, 4, 6 are not.

The first line contains the size of the list.

The rest of the lines contain the elements of the list.

Output a single integer value that represents the number of triples in the list.

In the example below, there are two triples: 4, 5, 6 and 5, 6, 7.

Exemple 1 :

Input :

```
6
1
2
4
5
6
7
```

Output :

```
2
```

Fichier de travail :

```
. \Learn_Kotlin_LdV\01-EXERCICES\16-Listes_mutables\16_4-
Boucles_for_sur_listes\P16_4_c-Triples
```

17. Programmation procédurale

17.1. Appel d'une fonction

- Exposé des concepts & savoirs :

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/4575>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

Problème 17_1-a : Pains

Enoncé :

Sam and Frodo decided to count how much bread they had. To do this, they created a function `totalLembas` that counts how much bread they have in total.

Your task is to invoke the function `totalLembas`; pass `breadFromFrodo` and `breadFromSam` as arguments.

Exemple 1 :

Input :

```
2  
3
```

Output :

```
5
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\17-Programmation_procedurale\17_1-  
Appel_fonction\P17_1-a-Pains
```

17.2. Déclaration d'une fonction

- Exposé des concepts & savoirs :

- Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/4580>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

Problème 17_2-a : Somme de 3 nombres

Enoncé :

Write a function named `sum` that takes three integer numbers and returns their sum.

Exemple 1 :

Input :

```
8  
11  
2
```

Output :

```
21
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\17-Programmation_procedurale\\17_2-
Declaration_fonction\\P17_2_a-Somme_3_nombres

Problème 17_2-b : Comparer deux sommes

Enoncé :

Write a function called `isGreater()` that takes 4 integer numbers and returns `true` if a sum of the first two arguments is greater than a sum of the third and fourth argument. Otherwise, return `false`.

Use the provided code template.

Exemple 1 :

Input :

```
1  
2  
3  
4
```

Output :

```
false
```

Exemple 2 :

Input :

```
3  
4  
1  
2
```

Output :

```
true
```

Exemple 3 :

Input :

```
2  
3  
4  
1
```

Output :

```
false
```

Fichier de travail :

`.\Learn_Kotlin_LdV\01-EXERCICES\17-Programmation_procedurale\17_2-
Declaration_fonction\P17_2_b-Comparer_2_sommes`

Problème 17_2-c : Voyelle ou pas ?

Enoncé :

Write a function that checks whether a letter of the basic Latin alphabet is a vowel. The input is one letter.

Do not consider the letter y a vowel.

Your function should take a `Char` type value and return a `Boolean`.

Use the provided code template.

Exemple 1 :

Input :

```
isVowel('a')
```

Output :

```
true
```

Exemple 2 :

Input :

```
isVowel('A')
```

Output :

```
true
```

Exemple 3 :

Input :

```
isVowel('b')
```

Output :

```
false
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\17-Programmation_procedurale\\17_2-
Declaration_fonction\\P17_2_c-Voyelle

17.3. Arguments par défaut

- Exposé des concepts & savoirs :
 - Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/4637>
- Validation de la compréhension par QCM :
 - Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER
- Mise en application :

Problème 17_3-a : Calculer le volume

Enoncé :

Create a function called `getVolume`, which takes 3 Int arguments that are supposed to be the length, the width, and the height of an object, and returns the volume of the cuboid with these sides. If the third or the second dimension is not specified, it should be 1 by default.

Exemple 1 :

Input :

```
getVolume(1)
```

Output :

```
1
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\17-Programmation_procedurale\\17_3-
Arguments_defaut\\P17_3_a-Calculer_volume

Problème 17_3-b : Gratuit

Enoncé :

Create a function called `tip` that calculates the tip. The function has two parameters `bill` and `percentage` and returns the amount of tip. The tip size varies greatly from country to country, so **10 %** of the total amount of a customer's check is a more or less universal solution.

So, if the percentage is not specified, set it at **10 %**.

Return only the integer part of gratuity.

Exemple 1 :

Input :

```
tip(100)
```

Output :

```
10
```

Exemple 2 :

Input :

```
tip(100, 5)
```

Output :

```
5
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\17-Programmation_procedurale\17_3-
Arguments_defaut\P17_3_b-Gratuit
```

Problème 17_3-c : Contrôle de vitesse

Enoncé :

Create a program that checks whether a car exceeds a speed limit.

Write a function `checkSpeed()` that will take two arguments: `speed` and `limit`. By default the limit is **60** kilometers per hour.

The speed contains the actual speed of the car. The limit contains the limit on the road. If there is no special restriction, you must apply the default limit.

Output :

- In case of excess write : "*Exceeds the limit by N kilometers per hour*" where **N** is kilometers per hour above the limit.
- Either write : "*Within the limit*".

Exemple 1 :

Input :

```
100  
60
```

Output :

```
Exceeds the limit by 40 kilometers per hour
```

Exemple 2 :

Input :

```
40  
90
```

Output :

```
Within the limit
```

Exemple 3 :

Input :

```
61  
no limit
```

Output :

```
Exceeds the limit by 1 kilometers per hour
```

Exemple 4 :

Input :

```
60  
no limit
```

Output :

Within the limit

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\17-Programmation_procedurale\\17_3-
Arguments_defaut\\P17_3_c-Contrôle_vitesse

17.4. Arguments nommés

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/4640>

- **Validation de la compréhension par QCM :**

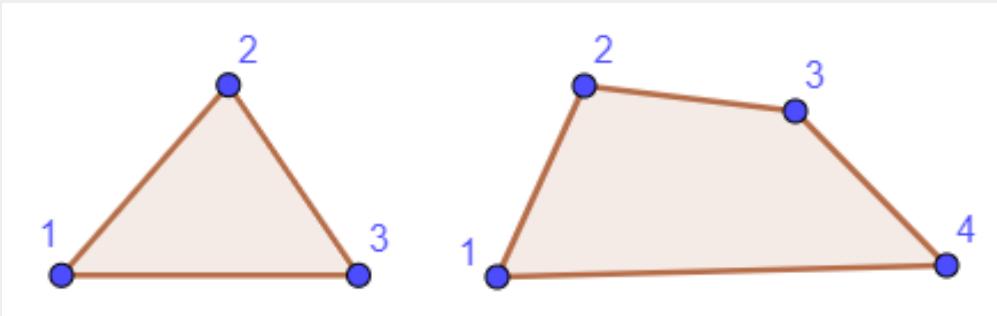
- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 17_4-a : Périmètre

Enoncé :

Change and implement the `perimeter()` function to calculate the perimeter of a polygon with 3 or 4 vertices, represented by X and Y coordinates. Vertices are passed sequentially.



You can use the `Math.hypot(..., ...)` function to calculate the length of the segment using the Pythagorean theorem. `Math.hypot(x, y)` returns the formula of the Pythagorean theorem is shown : $(x^2 + y^2)^{0.5}$

Exemple 1 :

Input :

```
perimeter(0.0, 0.0, 12.0, 0.0, 0.0, 5.0)
```

Output :

```
30.00
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\17-Programmation_procedurale\\17_4-
Arguments_nommes\\P17_4_a-Perimetre

Problème 17_4-b : URL**Enoncé :**

Write a function called `url` that creates a URL of a resource. The function has two parameters `host` and `port` and returns the URL in the following format : `https://HOST:PORT`.

Both host and port parameters are optional and have the following default values : `localhost` and `443` respectively.

Exemple 1 :*Input :*

```
url()
```

Output :

```
https://localhost:443
```

Exemple 2 :*Input :*

```
url("192.168.1.1", 2623)
```

Output :

```
https://192.168.1.1:2623
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\17-Programmation_procedurale\\17_4-
Arguments_nommes\\P17_4_a-URL

17.5. Portées des variables

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/17116>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'**ENT/Pronote** : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 17_5-a : Portée et puissance

Enoncé :

You have a program that prints the variable thirteen equal to 13 raised to the power of 2 to 10. There is a loop that raises the number 13 to a power at each iteration. Now the program prints only the result of exponentiation to the 2nd power.

Change the code below so that it prints the correct result (9 numbers, each on a new line).

Tip : You need to change the scope of the variable thirteen by moving one line of code.

Output :

```
169  
2197  
28561  
371293  
4826809  
62748517  
815730721  
10604499373  
137858491849
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\17-Programmation_procedurale\17_5-  
Portees_variables\P17_5_a-Portee_puissance
```

Problème 17_5-b : Cycliste

Enoncé :

While training, a cyclist rides the same road to and from. Now his distance controller is not working: there is an error in the controller code, and the road back is calculated incorrectly.

The variable `distance` contains the last distance traveled in one direction. If the cyclist is riding back, the `distance` variable must be negative. To calculate the total distance traveled, the negative `distance` must be inverted and then added to the `totalDistance` variable. However, for some reason, the code does not work now.

Fix the code to make it work.

Exemple 1 :

Input :

```
-10
```

30

Output :

40

Fichier de travail :

.\\Learn_Kotlin_Ldv\\01-EXERCICES\\17-Programmation_procedurale\\17_5-
Portees_variables\\P17_5_b-Cycliste

18. Les collections

18.1. Introduction aux collections

- **Exposé des concepts & savoirs :**
 - Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/10390>
- **Validation de la compréhension par QCM :**
 - Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER
- **Mise en application :**
 - Aucune application pour cette partie.

18.2. List (listes immuables)

- **Exposé des concepts & savoirs :**
 - Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/10730>
- **Validation de la compréhension par QCM :**
 - Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER
- **Mise en application :**

Problème 18_2-a : Nombres paires

Enoncé :

You are given a **List** of integers. Iterate through the given List and print in a single line the elements that are divisible by 2.

Exemple 1 :

Input :

8 11 13 2

Output :

8 2

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\18-Les_collections\\18_2-Listes\\P18_2_a-Nombres_paires

Problème 18_2-b : Somme de nombres

Enoncé :

Complete the `solution` function. You need to calculate and return the sum of numbers in the list `numbers`.

Exemple 1 :

Input :

3 2 15

Output :

20

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\18-Les_collections\\18_2-Listes\\P18_2_b-Somme_de_nombres

Problème 18_2-c : Nombre d'occurrences

Enoncé :

You are given a `List` of strings and a `String` in the input.

Calculate the number of occurrences of the `String` in the `List`.

Exemple 1 :

Input :

Fred Pola Mike Fred
Fred

Output :

2

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\18-Les_collections\\18_2-Listes\\P18_2_c-
Nombre_occurrences

Problème 18_2-d : Index

Enoncé :

In the input, you are given a **List** of products that you put on the shelf and a **String** that represents a specific product. Print out this product's position on the shelf. Note that products may repeat.

Exemple 1 :

Input :

```
Mustard Cheese Eggs Cola Eggs  
Eggs
```

Output :

```
2 4
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\18-Les_collections\\18_2-Listes\\P18_2_d-Index

18.3. Mutables List (listes mutables)

- Exposé des concepts & savoirs :

- Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/11032>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

Problème 18_3-a : Nouvelle liste

Enoncé :

You are given a **List** of integers and an **Int** in the input. Add the given the **List** with the **Integer** to a new **mutableList** and return the result.

Exemple 1 :

Input :

```
8 11 13 14  
1
```

Output :

```
8 11 13 14 1
```

Fichier de travail :

`.\\Learn_Kotlin_LdV\\01-EXERCICES\\18-Les_collections\\18_3-Listes_mutables\\P18_3_a-Nouvelle_liste`

Problème 18_3-b : Ajouter les tous

Enoncé :

Nous avons 2 listes `List` d'entiers en entrée. Retourner une `mutableList` comportant les 2 listes d'entrées.

Exemple 1 :

Input :

```
0 1 2 3  
4 5 6
```

Output :

```
0 1 2 3 4 5 6
```

Fichier de travail :

`.\\Learn_Kotlin_LdV\\01-EXERCICES\\18-Les_collections\\18_3-Listes_mutables\\P18_3_b-Ajouter_les_tous`

Problème 18_3-c : Supprime nous

Enoncé :

You are given a `MutableList` of `Strings` and an `Integer` in the input. Remove an element at the index of the specified integer.

Exemple 1 :

Input :

```
a b c d
```

Output :

```
a b d
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\18-Les_collections\18_3-Listes_mutables\P18_3_c-
Supprime_nous
```

Problème 18_3-d : Liste mutable

Enoncé :

We have a function `names`. When the function is called, a list `namesList: List<String>` is passed through the parameters. Implement counting the number of names in the input list.

Write the result of the calculation to the `nameCount` list in the following format : **The name Vasya is repeated 2 times.**

Exemple :

Input :

```
listOf("Vasya", "Olga", "Vasya", "Dima", "Masha", "Vasa", "Olga")
```

Output :

```
The name Vasya is repeated 2 times
The name Olga is repeated 2 times
The name Dima is repeated 1 times
The name Masha is repeated 1 times
The name Vasa is repeated 1 times
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\18-Les_collections\18_3-Listes_mutables\P18_3_d-
Liste Mutable
```

Problème 18_4-e : Banana

Enoncé :

You are given a `MutableList` of strings and a `String` in the input. Replace the occurrences of the given `String` in the `MutableList` with `Banana`.

Exemple 1 :

Input :

```
Sometimes you have to shake up your life  
shake
```

Output :

```
Sometimes you have to Banana up your life
```

Fichier de travail :

`.\Learn_Kotlin_LdV\01-EXERCICES\18-Les_collections\18_3-Listes_mutables\P18_3_e-Banana`

18.4. Maps (tableaux associatifs immuables)

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/12659>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'**ENT/Pronote** : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 18_4-a : Clé & valeur

Enoncé :

The `containsKeyAndValue` function gets map : `Map < String, String >` and the string `value: String` as input parameters.

If the map contains one or more `keys` that are equal to `value` and one or more `values` that are equal to `value`, the function returns `true`, otherwise it returns `false`.

Exemple 1 :

Input :

```
{one to two}, {three to one}, {four to five}  
one
```

Output :

```
true
```

Exemple 2 :

Input :

```
{10 to 20}, {30 to 1}, {55 to 23}
```

Output :

false

Fichier de travail :

.\Learn_Kotlin_LdV\01-EXERCICES\18-Les_collections\18_4-Maps\P18_4_a-Cle_et_valeur

Problème 18_4-b : Création d'une map

Enoncé :

You are given three **Strings** as input. Create a **Map** using these strings. Use the **String** as the key, and its length as the value. Return the resulting **Map**.

Exemple 1 :

Input :

I love Kotlin

Output :

{I to 1}, {love to 4}, {Kotlin to 6}

Exemple 2 :

Input :

Strings Ints Objects

Output :

{Strings to 7}, {Ints to 4}, {Objects to 7}

Fichier de travail :

.\Learn_Kotlin_LdV\01-EXERCICES\18-Les_collections\18_4-Maps\P18_4_b-Creation_map

Problème 18_4-c : Somme de valeurs

Enoncé :

You are given **Map<Int, Int>** as input. Return the sum of all values that have even keys.

Exemple 1 :

Input :

```
{100 to 10}, {55 to 3}, {112 to 5}
```

Output :

```
15
```

Exemple 2 :

Input :

Output :

```
5
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\18-Les_collections\18_4-Maps\P18_4_c-Somme_valeurs
```

Problème 18_4-d : Application

Enoncé :

In the store, all products are stored in a `Map<String, Int>`, which holds name - price pairs. The customer comes with a shopping list and wants to know what the total price of products on the list will be. Keep in mind that some products may not be available in the store.

You are given `Map<String, Int>` (product name and price) and `MutableList` of `String` (shopping list) as the function parameters. Return the total price of the items the customer needs to buy.

Exemple 1 :

Input :

```
{Cola to 500}, {Apple to 1500}, {Banana to 300}  
Cola Apple
```

Output :

```
2000
```

Exemple 2 :

Input :

```
{Sprite to 150}, {Lays to 200}, {Milk to 600}, {Snickers to 100}  
Sprite Lays Coffee
```

Output :

```
350
```

Fichier de travail :

`.\\Learn_Kotlin_LdV\\01-EXERCICES\\18-Les_collections\\18_4-Maps\\P18_4_d-Application`

18.5. Mutable Map (tableaux associatifs mutables)

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/15876>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 18_5-a : Notes aux examens

Enoncé :

Write a program that reads the names of students and their exam scores, adds them to `MutableMap`, and prints the result. In this `MutableMap`, names are keys, and scores are values. It is guaranteed that the keys are `Strings` and the values are of `Int` type.

Keys and values must be read on a new line each. Input is over when the program receives the word "stop" instead of the next key. If the same key is entered more than one time, `MutableMap` must keep the value that was entered first.

Input : key :* `String`, value: `Int` (on a new line), until the word "stop" is read.

Output : resulting `MutableMap<String, Int>`

Exemple 1 :

Input :

```
Alice  
88  
John  
92  
Harry  
60
```

```
stop
```

Output :

```
{Alice=88, John=92, Harry=60}
```

Exemple 2 :

Input :

```
Alice  
45  
Alice  
66  
stop
```

Output :

```
{Alice=45}
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\18-Les_collections\18_5-Maps_mutables\P18_5_a-  
Notes_examens
```

Problème 18_5-b : Liste de souhaits

Enoncé :

Imagine that you have a wish list. It's very long and you want all the listed items; however, you can't buy the items with the price greater than **limit**.

Write a function **makeMyWishList** that returns a map with all the elements with the price lower than or equal to **limit**.

Exemple 1 :

Input :

```
mapOf("watch" to 120, "jumper" to 80, "smartphone" to 350)  
200
```

Output :

```
{watch=120, jumper=80}
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\18-Les_collections\\18_5-Maps_mutables\\P18_5_b-
Liste_de_souhaits

18.6. Set (Sets immuables)

- Exposé des concepts & savoirs :

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/11290>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

Problème 18_6-a : Combinaison

Enoncé :

You are given two sets of strings in the input. Combine them and return the result.

Exemple 1 :

Input :

```
Hello how  
Hello are you
```

Output :

```
Hello how are you
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\18-Les_collections\\18_6-Sets\\P18_6_a-Combinaison

Problème 18_6-b : Conversion

Enoncé :

Given a `MutableList` of `Strings` in the input, convert it to `Set`.

Exemple 1 :

Input :

```
A B C D A
```

Output :

```
A B C D
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\18-Les_collections\18_6-Sets\P18_6_b-Conversion
```

Problème 18_6-c : Soustraction

Enoncé :

You are given two sets of integers in the input. Subtract one from the other and return the size of the resulting set.

Exemple 1 :

Input :

```
8 11 12 13  
8 12
```

Output :

```
2
```

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\18-Les_collections\18_6-Sets\P18_6_c-Soustraction
```

18.7. Mutable Set (Sets mutables)

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/11897>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 18_7-a : Combinaison

Enoncé :

You are given a **Set** and a **MutableList** of strings as input. Combine them into a **MutableSet** and return the result.

Exemple 1 :

Input :

```
A B C  
A B B D C
```

Output :

```
A B C D
```

Fichier de travail :

`.\Learn_Kotlin_LdV\01-EXERCICES\18-Les_collections\18_7-Sets_mutables\P18_7_a-Combinaison`

Problème 18_7-b : Soustraction

Enoncé :

You are given a **Set** of strings and a **String** as input. Remove the given string from the **Set** and return the result.

Exemple 1 :

Input :

```
Hello it's you me  
you
```

Output :

```
Hello it's me
```

Fichier de travail :

`.\Learn_Kotlin_LdV\01-EXERCICES\18-Les_collections\18_7-Sets_mutables\P18_7_b-Soustraction`

Problème 18_7-c : filtre

Enoncé :

Complete the `evenFilter()` function that takes a **Set** of **Int** and returns a new **Set** containing only the even elements from that set. Don't use the `+=` operator: work with `MutableSet` because it is faster than reassigning.

Exemple 1 :

Input :

```
1 2 3 4 5 6 7 8
```

Output :

```
2 4 6 8
```

Fichier de travail :

`.\\Learn_Kotlin_LdV\\01-EXERCICES\\18-Les_collections\\18_7-Sets_mutables\\P18_7_c-Filtre`

19. La Programmation Orientée Objet (P.O.O.)

19.1. Déclaration d'une classe

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/6200>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 19_1-a : Nombres complexes

Enoncé :

Here's an idea. How about writing a calculator that manipulates complex numbers ?

Where to start ? Write a class named `Complex`. It must have two `Double` properties named `real` and `image`.

The properties must be `assignable` and their default value must be `zero`.

Fichier de travail :

`.\\Learn_Kotlin_LdV\\01-EXERCICES\\19-P00\\19_1-Declaration_classe\\P19_1_a-Nombres_complexes`

Problème 19_1-b : Nouvelle planète

Enoncé :

Create the class `Planet` with the following properties :

- `star` : type `String`, with the default value "Unknown Star";

- **mass** : type `Double`, with the default value **1.0**;
- **numberOfSatellites** : type `Int`, with the default value **1**.

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\19-P00\19_1-Declaration_classe\P19_1_b-Planete
```

Problème 19_1-c : Surface

Enoncé :

Assume you have the `Rectangle` class with `width` and `height` properties. The properties types are `Int`.

Implement a function that receives a `Rectangle` object and prints the product of the multiplication of two properties. Name the function `printArea`.

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\19-P00\19_1-Declaration_classe\P19_1_c-Surface
```

19.2. Constructeurs (par défaut et primaire)

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/10740>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 19_2-a : Compte courant

Enoncé :

Implement the `BankAccount` class which receives two `Longs` called `depositedv` and `withdrawn` in the constructor and sets these properties :

- `deposited` (the copy),
- `withdrawn` (the copy),
- `balance` (the difference).

Fichier de travail :

```
.\Learn_Kotlin_LdV\01-EXERCICES\19-P00\19_2-Constructeurs_defaut_et_primaire\P19_2_a-Compte_courant
```

Problème 19_2-b : Utilisation d'Init

Enoncé :

There is a program that reads a number. If the number is less than **-128**, then the property `time` of a `ByteTimer` class must be **-128**. If it's greater than **127**, then it must be **127**, otherwise, it must be its raw value.

Fix the `ByteTimer` class.

Tip : Use an init block.

Exemple 1 :

Input :

-200

Output :

-128

Fichier de travail :

`.\Learn_Kotlin_LdV\01-EXERCICES\19-P00\19_2-Constructeurs_defaut_et_primaire\P19_2_b-utilisation_init`

19.3. Fonctions membres (méthodes)

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/6364>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 19_3-a : Bug dans VinylStore

Enoncé :

You have a class `VinylStore` with one member function `printVinyl()`, which should print the name of the vinyl, but now it works incorrectly.+

Find and fix the bugs.

Fichier de travail :

`.\Learn_Kotlin_LdV\01-EXERCICES\19-P00\19_3-Methodes\P19_3_a-Bug_VinylStore`

Problème 19_3-b : Volume d'un cube

Enoncé :

Given a class named `Box`. It has three `Double` properties.

Write a member function to calculate the volume of the box. The function must be named as `getVolume`. It should take no arguments and return a `Double` result.

Fichier de travail :

`.\Learn_Kotlin_LdV\01-EXERCICES\19-P00\19_3-Methodes\P19_3_b-Cube_volume`

Problème 19_3-c : Automobile

Enoncé :

Given a class named `Car` containing three properties :

- the `Int` property `year`,
- the `String` property `make`, *the `Int` property `speed`.

Add the following member functions to the class :

- `accelerate()` : adds 5 to the speed property each time it's called;
- `decelerate()` : subtracts 5 from the speed property each time it's called. The speed cannot be less than zero, so if you get a negative speed, just make it equal to 0.

Fichier de travail :

`.\Learn_Kotlin_LdV\01-EXERCICES\19-P00\19_3-Methodes\P19_3_c-Automobile`

19.4. Accesseurs & Mutateurs

- Exposé des concepts & savoirs :

- Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/10746>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

Problème 19_4-a : Population d'une ville

Enoncé :

Imagine we want to implement the `City` class which stores the `name` of the city and its `population`.

There are some restrictions. If the number is out of range, the `population` is corrected like

this: zero for negative values and 50 million for greater than 50_000_000.

Finish the implementation of the `City` class following this logic.

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\19-POO\\19_4-Accessseurs_et_mutateurs\\P19_4_a-Ville

Problème 19_4-b : Classe LewisCarrollBook

Enoncé :

Write a class `LewisCarrollBook` with custom getters and setters, which has three fields :

- `name`: with the default value "",
- `author`: with the default value "Lewis Carroll",
- `price` : with the default value 0.

main() fonction :

```
fun main() {  
    val lewisCarrollBook = LewisCarrollBook()  
    var bookName = lewisCarrollBook.name  
    lewisCarrollBook.name = "Alice's Adventures in Wonderland"  
    bookName = lewisCarrollBook.name  
    var authorName = lewisCarrollBook.author  
    var bookPrice = lewisCarrollBook.price  
    lewisCarrollBook.price = 1500  
}
```

Output :

```
The name of the book is  
Now, a book called Alice's Adventures in Wonderland  
The name of the book is Alice's Adventures in Wonderland  
The author of the book is Lewis Carroll  
Putting a new price...  
The new price is 1500
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\19-POO\\19_4-Accessseurs_et_mutateurs\\P19_4_b-Classe_LewisCarrollBook

19.5. Constructeurs secondaires

• Exposé des concepts & savoirs :

- Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/10746>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

Problème 19_5-a : Kitty

Enoncé :

You went to the "Cat House" looking for the cutest kitty.

The class **Kitty** has one **String** property **color** and one **Int** property **age**. Write all possible secondary constructors.



- first → (**String**, **Int**)
- second → (**Int**, **String**)
- third - > (**String**)
- last → (**Int**)

Fichier de travail :

. \Learn_Kotlin_LdV\01-EXERCICES\19-P00\19_5-Constructeurs_secondaires\P19_5_a-Kitty

20. Les tableaux

20.1. Le type **Array**

- Exposé des concepts & savoirs :

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/4567>

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

Problème 20_1-a : Initialiser un tableau de caractères

Enoncé :

Create an array of chars named **characters** with four elements : **a**, **z**, **e**, and **d**.
After that, please, output it.

Use the provided code template.

Output :

```
a, z, e, d
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\20-Les_tableaux\\20_1-Arrays\\P20_1-a-Init_array

Problème 20_1-b : Echanger

Enoncé :

Suppose you have a numbers array of the `IntArray` type.

Swap its first and last elements.

It is guaranteed that the size of the array is larger than 0.

Input :

```
1 2 3 4 5 6
```

Output :

```
6 2 3 4 5 1
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\20-Les_tableaux\\20_1-Arrays\\P20_1-b-Echanger

20.2. Les tableaux de Strings

- Exposé des concepts & savoirs :

- Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/11477>



Nous allons principalement travailler avec des Arrays de Strings dans cette partie. Mais les Arrays peuvent ici prendre comme valeur n'importe quel type/classe d'objets.

- Validation de la compréhension par QCM :

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- Mise en application :

Problème 20_2-a : Addition de tableaux

Enoncé :

Concatenate two arrays `firstArray` and `secondArray` and print the result.

In the example below, each line corresponds to a separate array. Elements are separated by spaces.

Tip : Use the function `joinToString()`.

Exemple 1 :

Input :

```
valar morghulis  
valar dohaeris
```

Output :

```
valar, morghulis, valar, dohaeris
```

Fichier de travail :

`.\Learn_Kotlin_LdV\01-EXERCICES\20-Les_tableaux\20_2-String_arrays\P20_2_a-Addition`

20.3. Les tableaux à N dimensions

- **Exposé des concepts & savoirs :**
 - Lire le contenu proposé par JetBrains Academy : <https://hyperskill.org/learn/step/4564>
- **Validation de la compréhension par QCM :**
 - Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER
- **Mise en application :**

Problème 20_3-a : Coins

Enoncé :

You already have a 2D array `inputArray`. Print all of its corners in the following order : left to right and top to bottom.

Print the result for two elements in the line. Use a single space to separate two elements on the same line.

Take a look at the examples below, and remember that nested arrays can be of different lengths !

Tip : You can use `first()` and `last()`

Exemple 1 :

Input :

```
3  
1 2 3  
4 5 6  
7 8 9
```

Output :

```
1 3  
7 9
```

Fichier de travail :

.\\Learn_Kotlin_LdV\\01-EXERCICES\\20-Les_tableaux\\P20_3_a-Coins

20.4. Boucle **for** sur les tableaux

- **Exposé des concepts & savoirs :**

- Lire le contenu proposé par **JetBrains Academy** : <https://hyperskill.org/learn/step/XXXX>

- **Validation de la compréhension par QCM :**

- Réaliser le QCM de l'ENT/Pronote : LIEN_QCM_A_DEPOSER

- **Mise en application :**

Problème 20_4-a : Triples

Enoncé :

Write a program that reads an array of integers and outputs the number of triples in the array.

A triple is three consecutive integers in ascending order like : 3, 4, 5 is a triple, but 5, 4, 3 and 2, 4, 6 are not.

The first line contains the size of the array.

The rest of the lines contain elements of the array.

Output a single integer value that is the number of triples in the array.

In the example below, there are two triples: 4, 5, 6 and 5, 6, 7.

Exemple 1 :

Input :

```
6  
1  
2  
4  
5  
6  
7
```

Output :

2

Fichier de travail :

`.\Learn_Kotlin_LdV\01-EXERCICES\20-Les_tableaux\20_4-Boucles_for_sur_tableaux\P20_4-a-Triples`