**Atelier 1 : Introduction au Java**

1. **Définition**

Java est un langage de programmation populaire, créé en 1995. Il appartient à Oracle.

Il est utilisé pour:

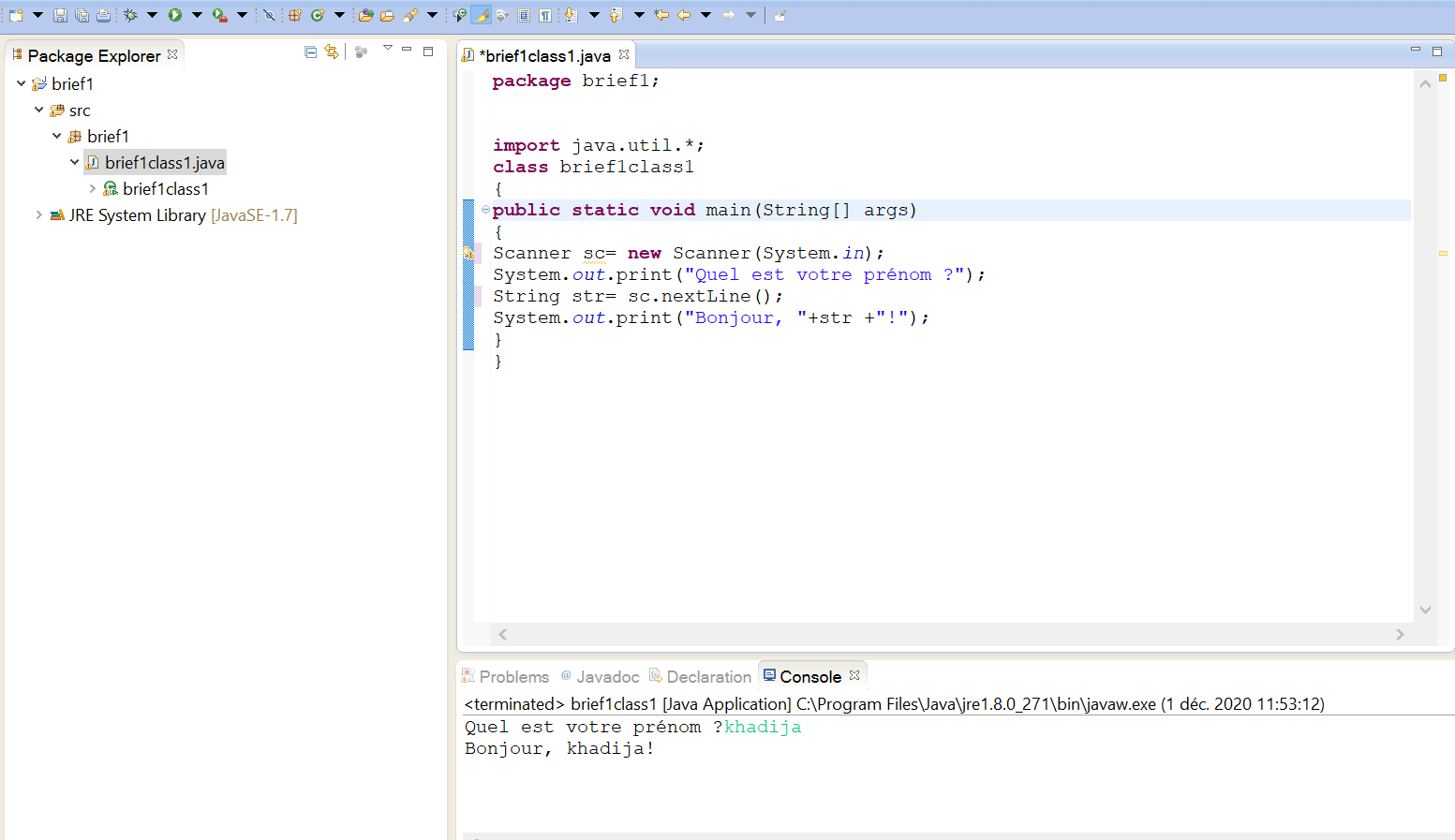
* Applications mobiles (en particulier les applications Android)
* Applications de bureau
* des applications Web
* Serveurs Web et serveurs d'applications
* Jeux
* Connexion à la base de données
* …

Pourquoi utiliser Java?

* Java fonctionne sur différentes plates-formes (Windows, Mac, Linux, Raspberry Pi, etc.)
* C'est l'un des langages de programmation les plus populaires au monde
* Il est facile à apprendre et simple à utiliser
* Il est open-source et gratuit
* Il est sécurisé, rapide et puissant
* Il a un énorme soutien de la communauté (des dizaines de millions de développeurs)
* Java est un langage orienté objet qui donne une structure claire aux programmes et permet de réutiliser le code, réduisant ainsi les coûts de développement

1. **Les variables**
2. Créer une variable, et demander à l'utilisateur d’entrer son prénom. La machine répond par un charmant « Bonjour » suivi du prénom. On aura ainsi le dialogue suivant :

Machine : Quel est votre prénom ?  
Utilisateur : Bouchra  
Machine : Bonjour, Bouchra !



Chaque ligne de code exécutée en Java doit se trouver dans un fichier **class**. Dans notre exemple, nous avons nommé la classe brief1class1. Une classe doit toujours commencer par une première lettre majuscule parce que java est « case sensitive ».

# \***main() méthode**, Tout code à l'intérieur de la main()méthode sera exécuté.

# \* **System.out.println ()** : méthode pour imprimer une ligne de texte à l'écran

# \* **variables** sont des élément utiliser pour stocker les donnes et pour créer une variable, vous devez spécifier le type et lui attribuer une valeur : type variable = value;

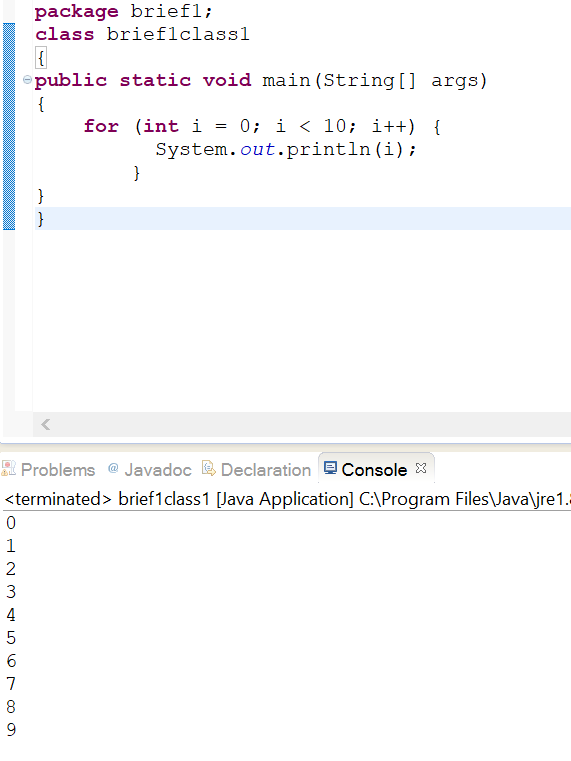
1. Une société de produit cosmétique souhaitera créer une calculatrice de TVA avec remise.

L'utilisateur saisit un montant HT et s'il le souhaite une remise, le résultat TTC s'affiche sur la console.

* L'utilisateur doit pouvoir répondre Oui ou Yes à la demande de remise.
* La saisie de la remise se fait en pourcentage, un nombre à virgule donc, et elle s'applique sur le montant HT autrement il y a fraude à la TVA ;-)
* Si une remise est appliquée, il faut à la fin réafficher le pourcentage de remise.
* S'il n'y a pas de remise il faut à la fin afficher "Aucune remise n'a été appliquée"
* Il faut répéter le moins de code possible, notamment il ne faut pas répéter le calcul du montant TTC final.

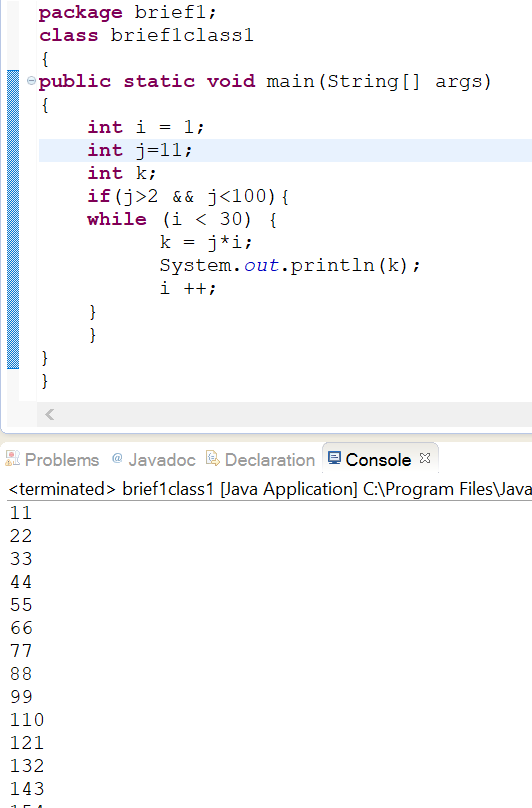
1. **Les boucles**
2. Créer une variable et l'initialiser à 0. Tant que cette variable n'atteint pas 10 :

* Afficher
* Incrémenter de 1



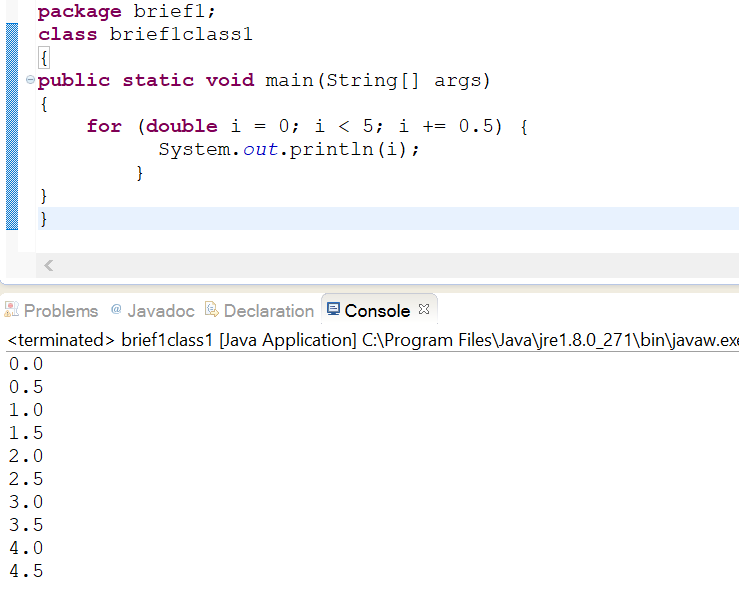
1. Créer deux variables. Initialiser la première à 1 et la deuxième avec un nombre compris en 2 et 100. Tant que la première variable n'est pas supérieure à 30 :

* Multiplier la première variable avec la deuxième
* Afficher le résultat
* Incrémenter la première variable

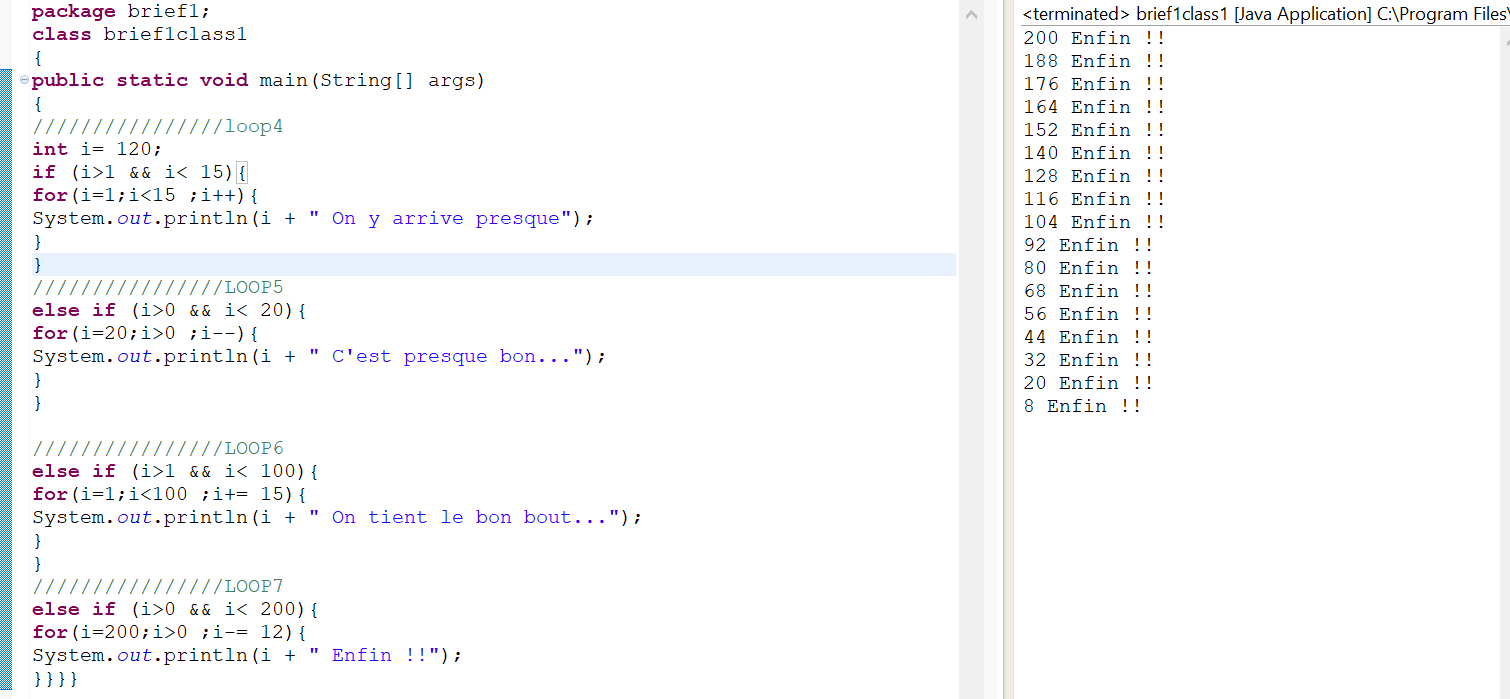


1. Créer une variable et l'initialiser à 1. Tant que cette variable n'atteint pas 10 :

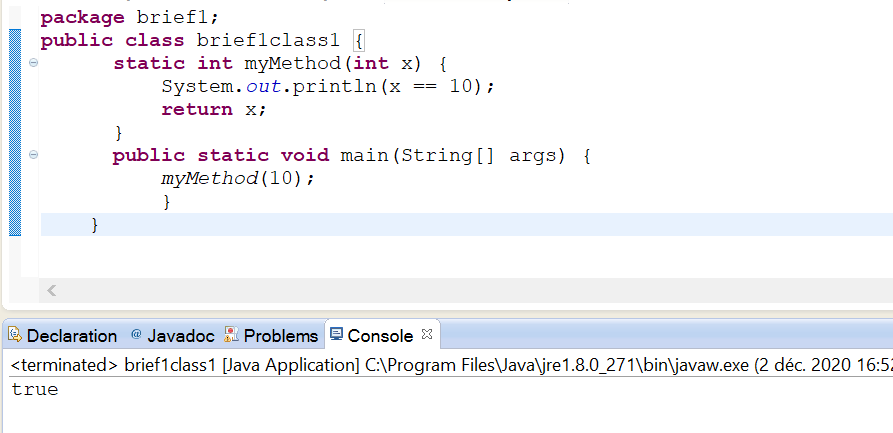
* Afficher
* Incrémenter de la moitié de sa valeur

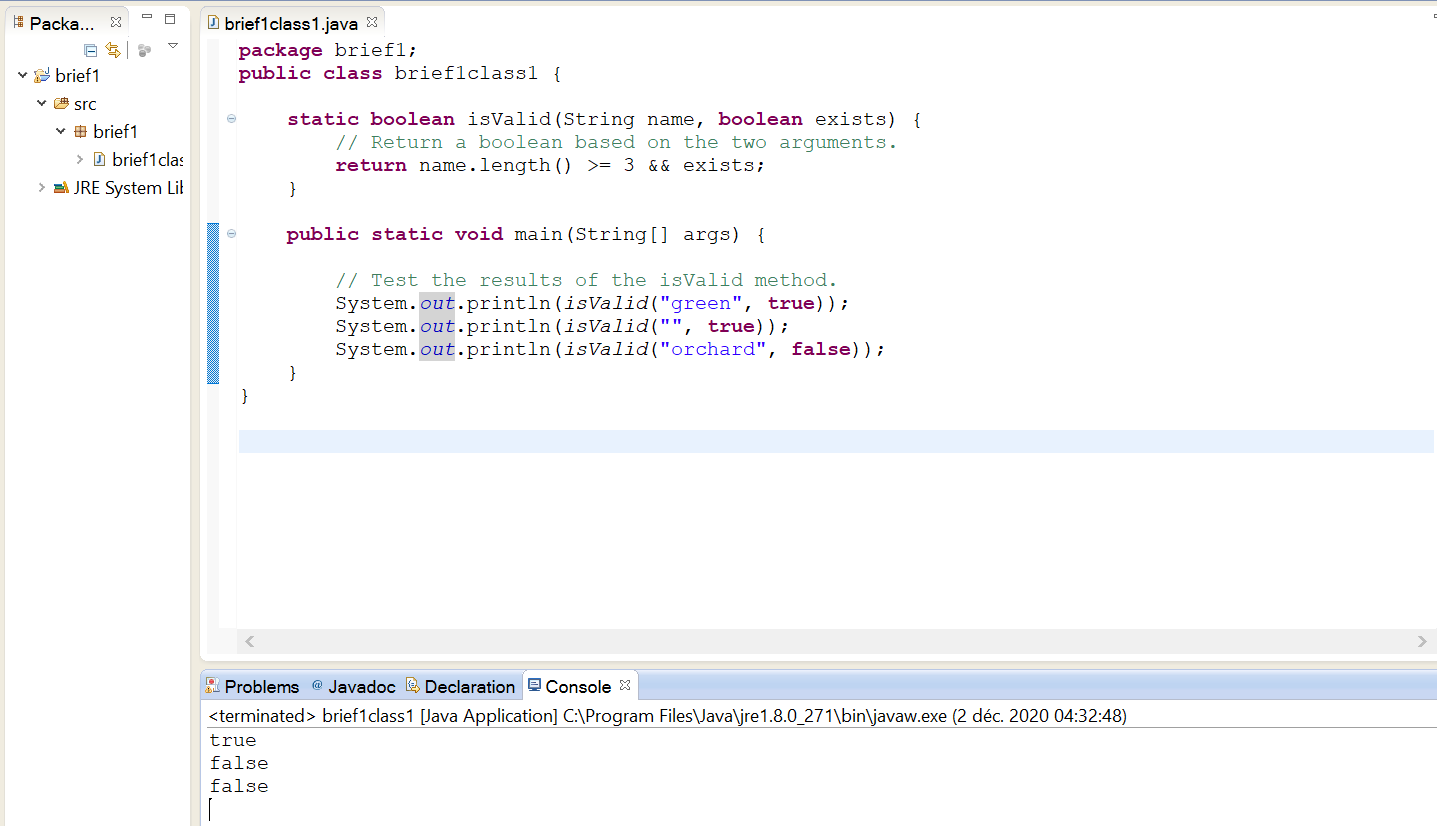


1. En allant de 1 à 15 avec un pas de 1, afficher le message On y arrive presque...
2. En allant de 20 à 0 avec un pas de 1, afficher le message ‘C'est presque bon...’
3. En allant de 1 à 100 avec un pas de 15, afficher le message ‘On tient le bon bout...’
4. En allant de 200 à 0 avec un pas de 12, afficher le message ‘Enfin ! ! !’

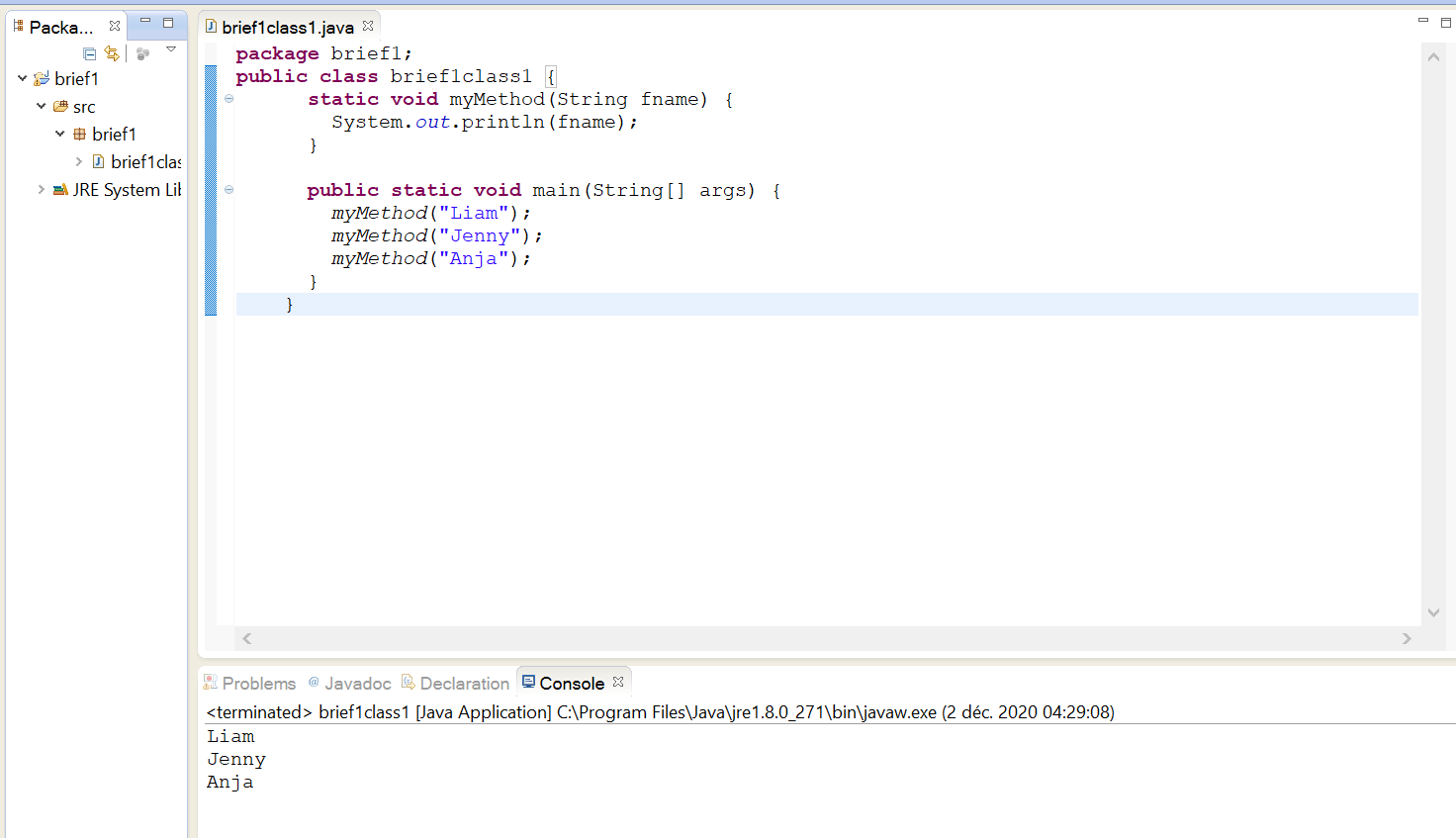


1. **Les fonctions**
2. Faire une fonction qui retourne True.





1. Faire une fonction qui prend en paramètre une chaine de caractères et qui retourne cette même chaine.



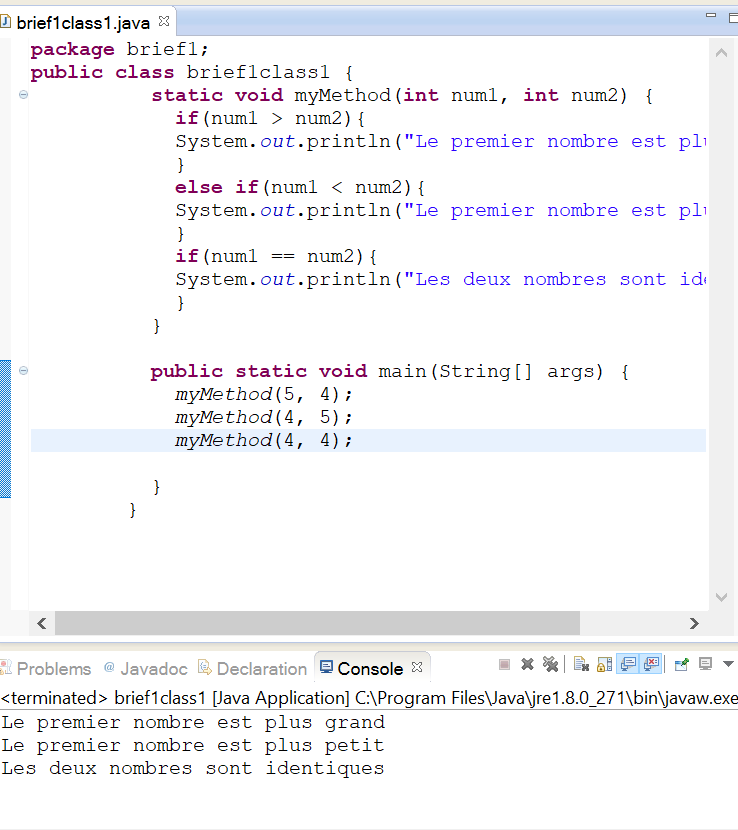
* myMethod() est le nom de la méthode
* static signifie que la méthode appartient à la classe Main et non à un objet de la classe Main.
* Void signifie que cette méthode n'a pas de valeur de retour.

1. Faire une fonction qui prend en paramètre deux chaines de caractères et qui renvoi la concaténation de ces deux chaines.

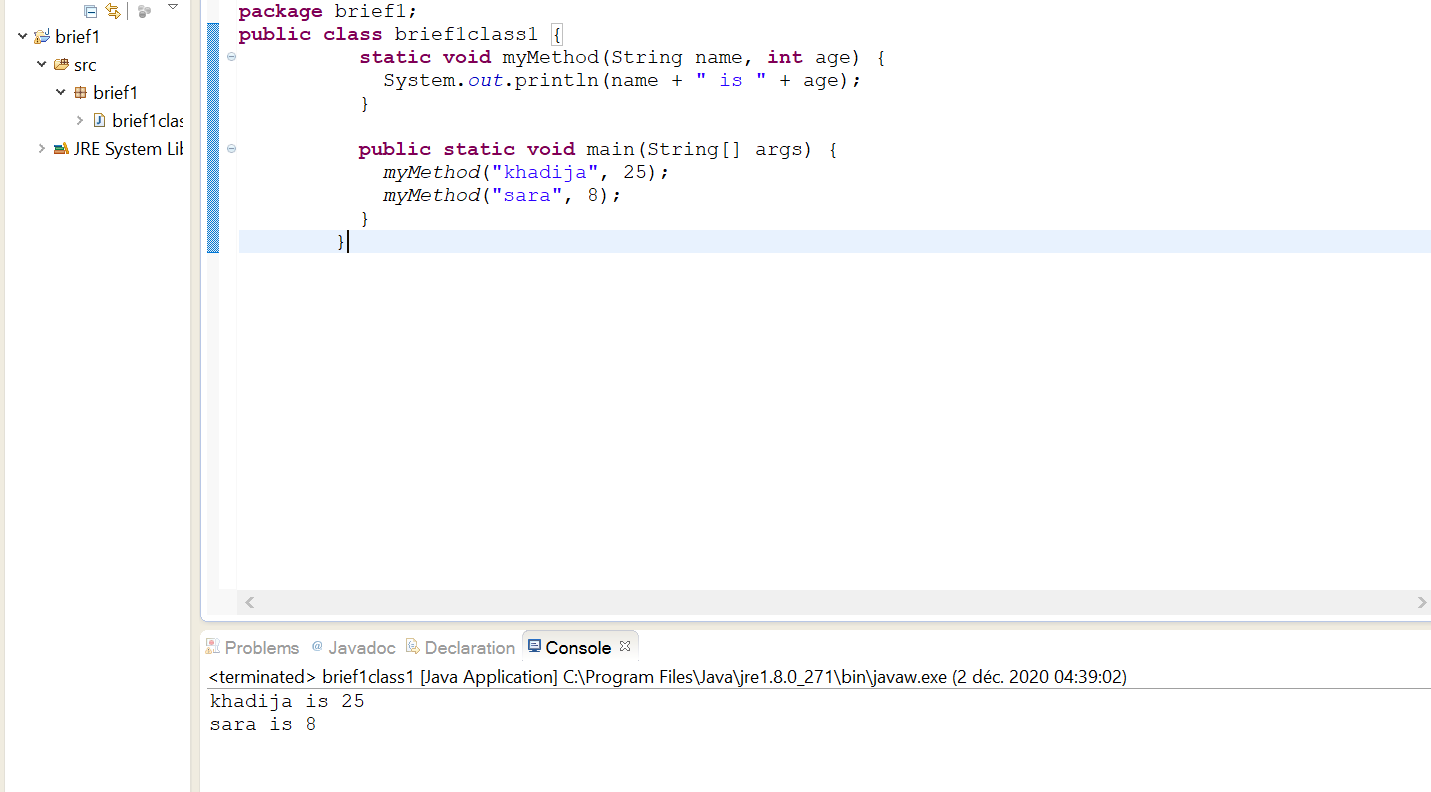


1. Faire une fonction qui prend en paramètre deux nombres. La fonction doit retourner :

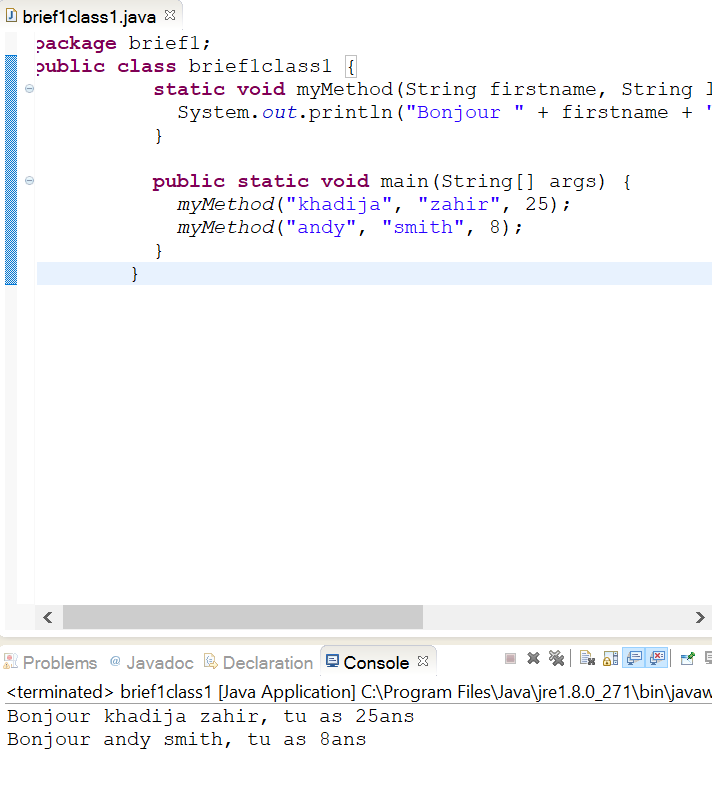
* Le premier nombre est plus grand si le premier nombre est plus grand que le deuxième
* Le premier nombre est plus petit si le premier nombre est plus petit que le deuxième
* Les deux nombres sont identiques si les deux nombres sont égaux



1. Faire une fonction qui prend en paramètre un nombre et une chaine de caractères et qui renvoi la concaténation de ces deux paramètres.

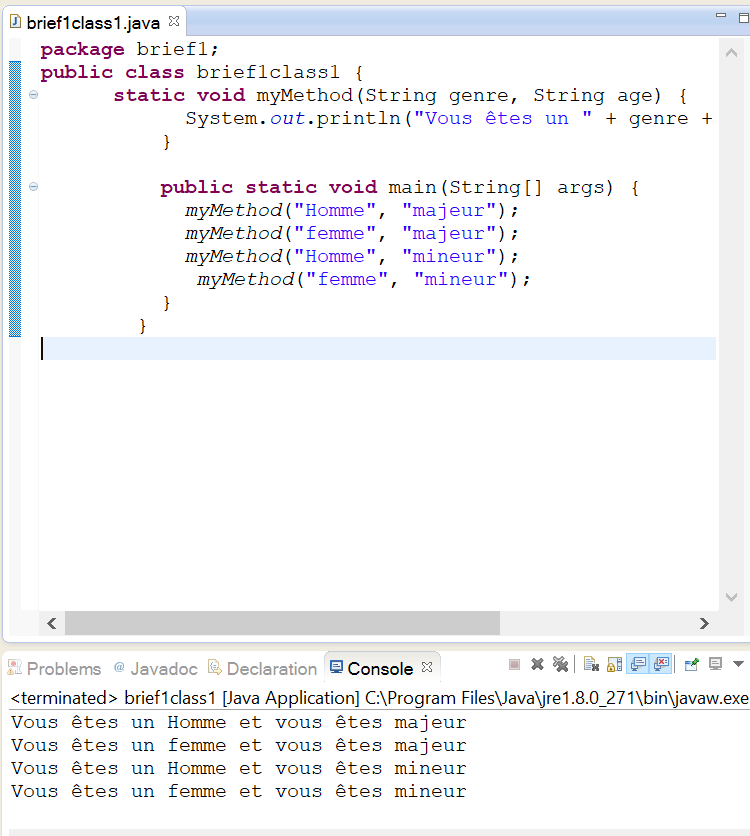


1. Faire une fonction qui prend trois paramètres : nom, prenom et age. Elle doit renvoyer une chaine de la forme : "Bonjour" + nom + prenom + ", tu as " + age + "ans".

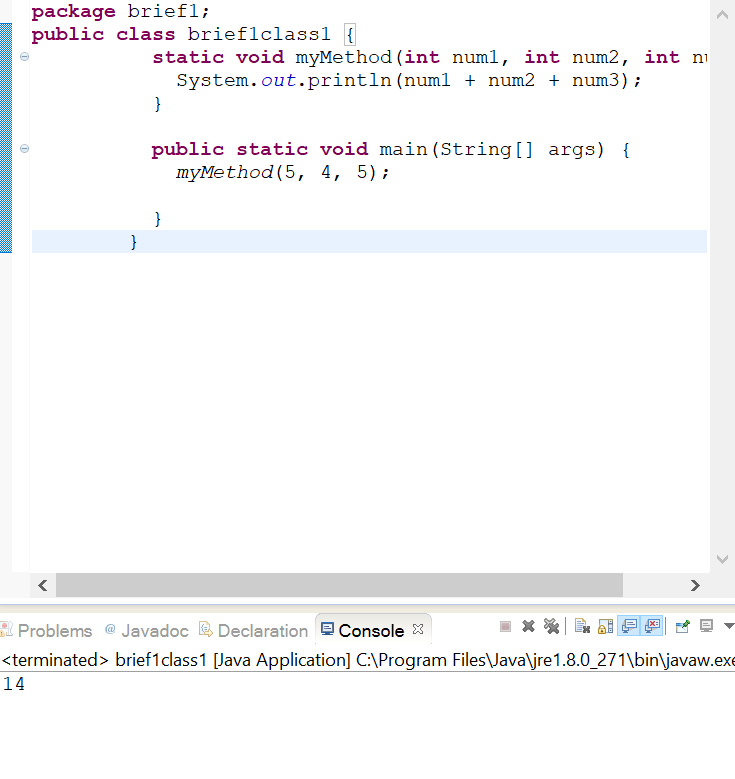


1. Faire une fonction qui prend deux paramètres : age et genre. Le paramètre genre peut prendre comme valeur Homme ou Femme. La fonction doit renvoyer en fonction des paramètres (gérer tous les cas) :

* Vous êtes un homme et vous êtes majeur
* Vous êtes un homme et vous êtes mineur
* Vous êtes une femme et vous êtes majeur
* Vous êtes une femme et vous êtes mineur



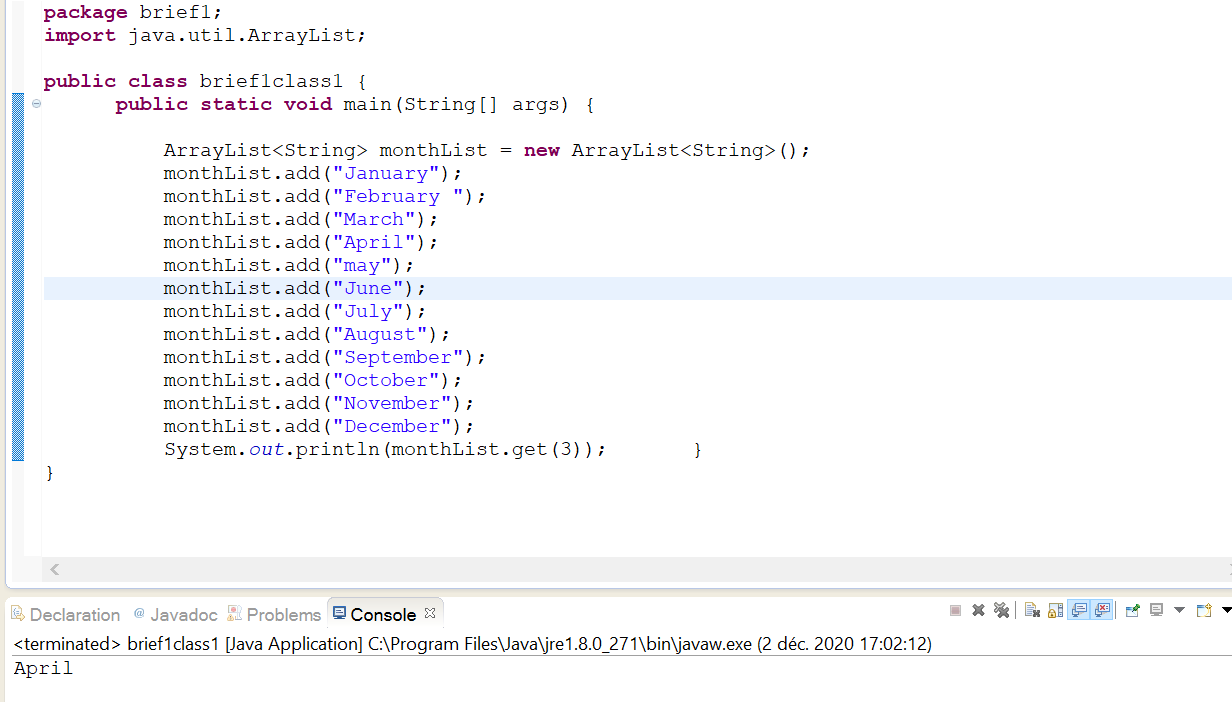
1. Faire une fonction qui prend en paramètre trois nombres et qui renvoi la somme de ces nombres. Tous les paramètres doivent avoir une valeur par défaut.



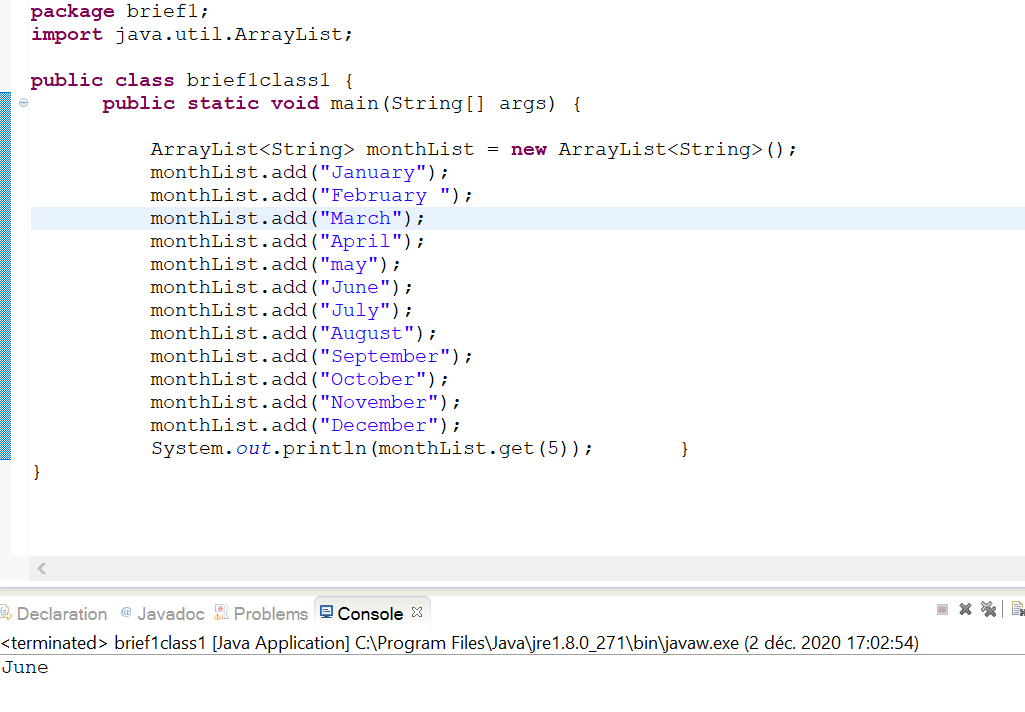
1. **Les collections**
2. Créer une liste de mois, ajouter le nom des douze mois de l'année et afficher la collection.



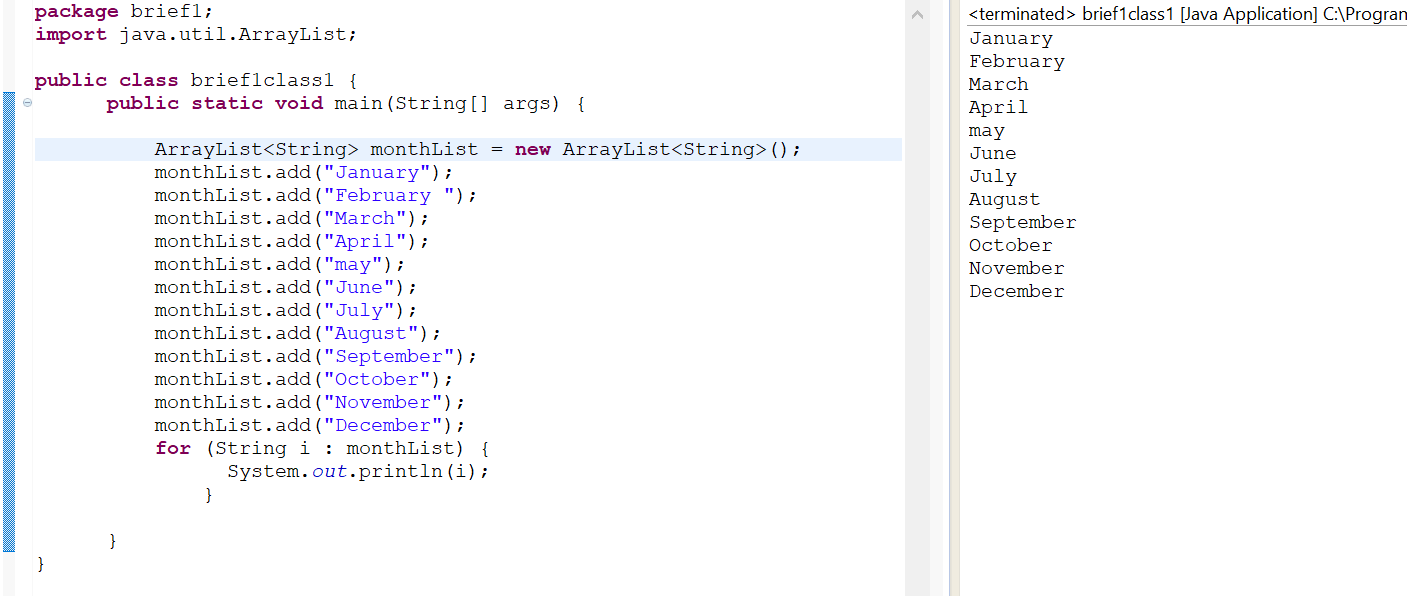
1. Afficher la valeur de la troisième ligne de cette liste (question 1).



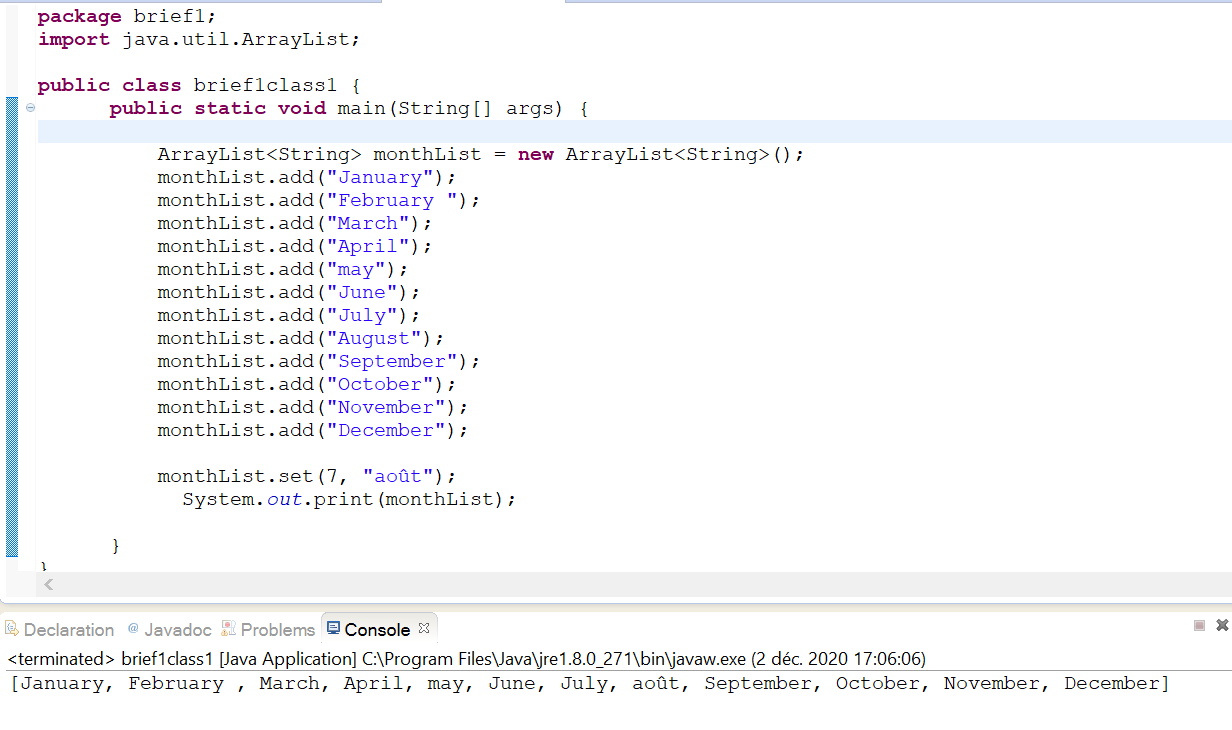
1. Avec la liste de l'exercice, afficher la valeur de l'index 5.



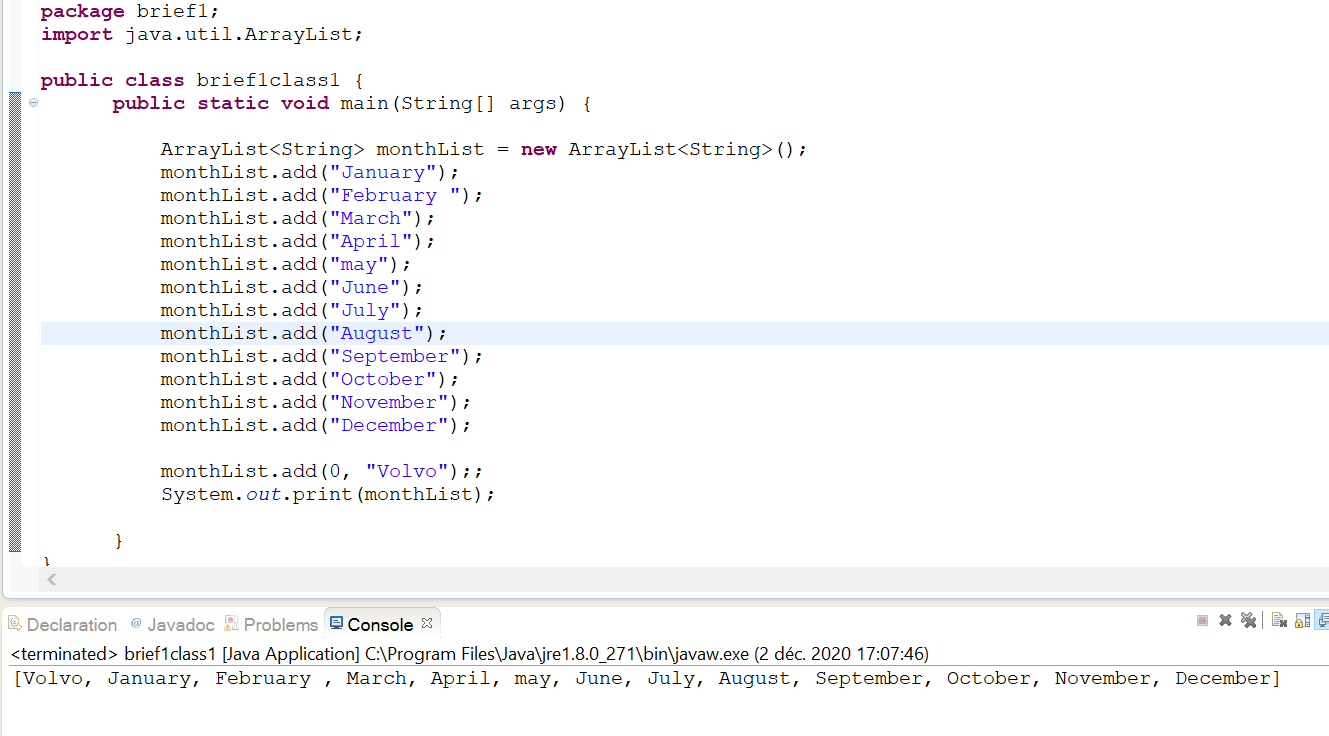
1. Parcourir tous les éléments de cette liste



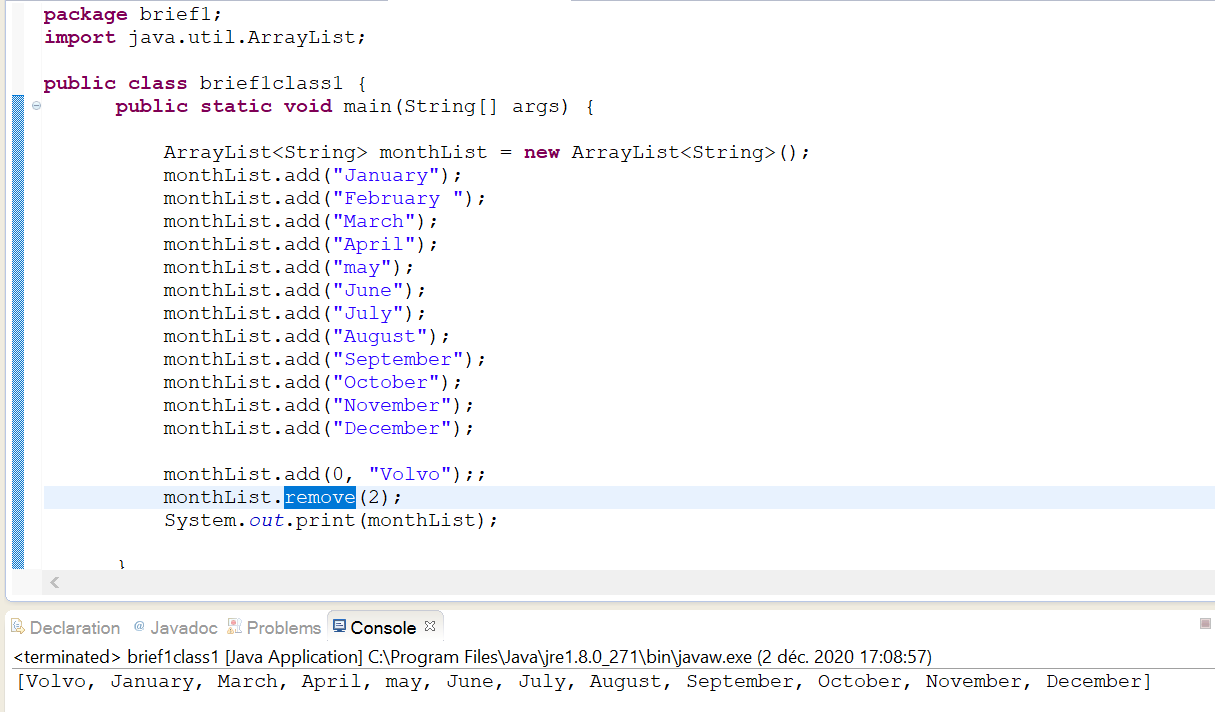
1. Avec la liste de l'exercice, modifier le mois de aout pour lui ajouter l'accent manquant.



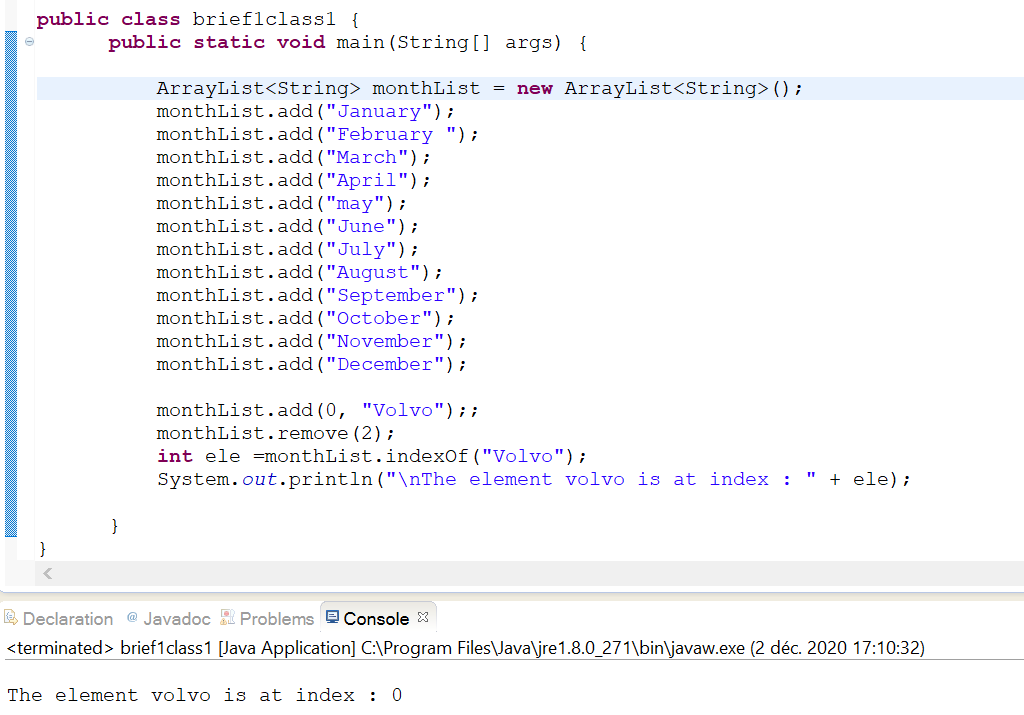
1. Insérer un élément dans la liste à la première position.



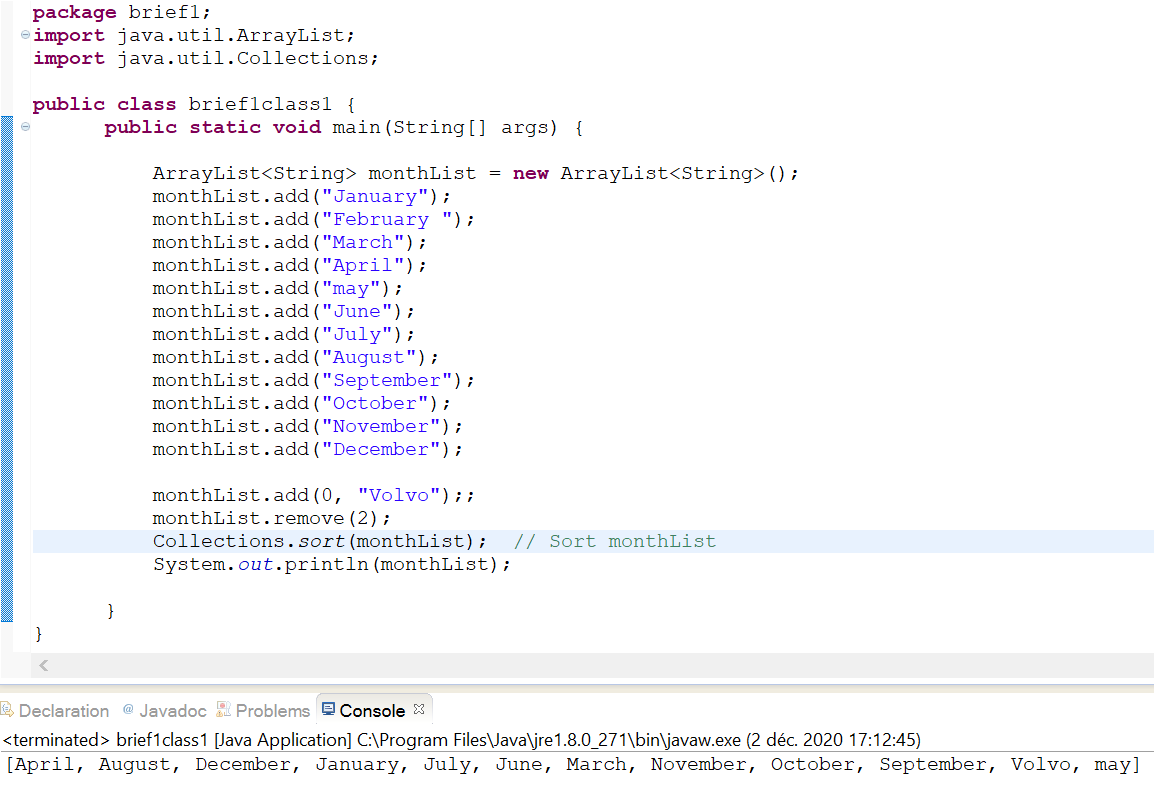
1. Supprimer le troisième élément de cette liste.



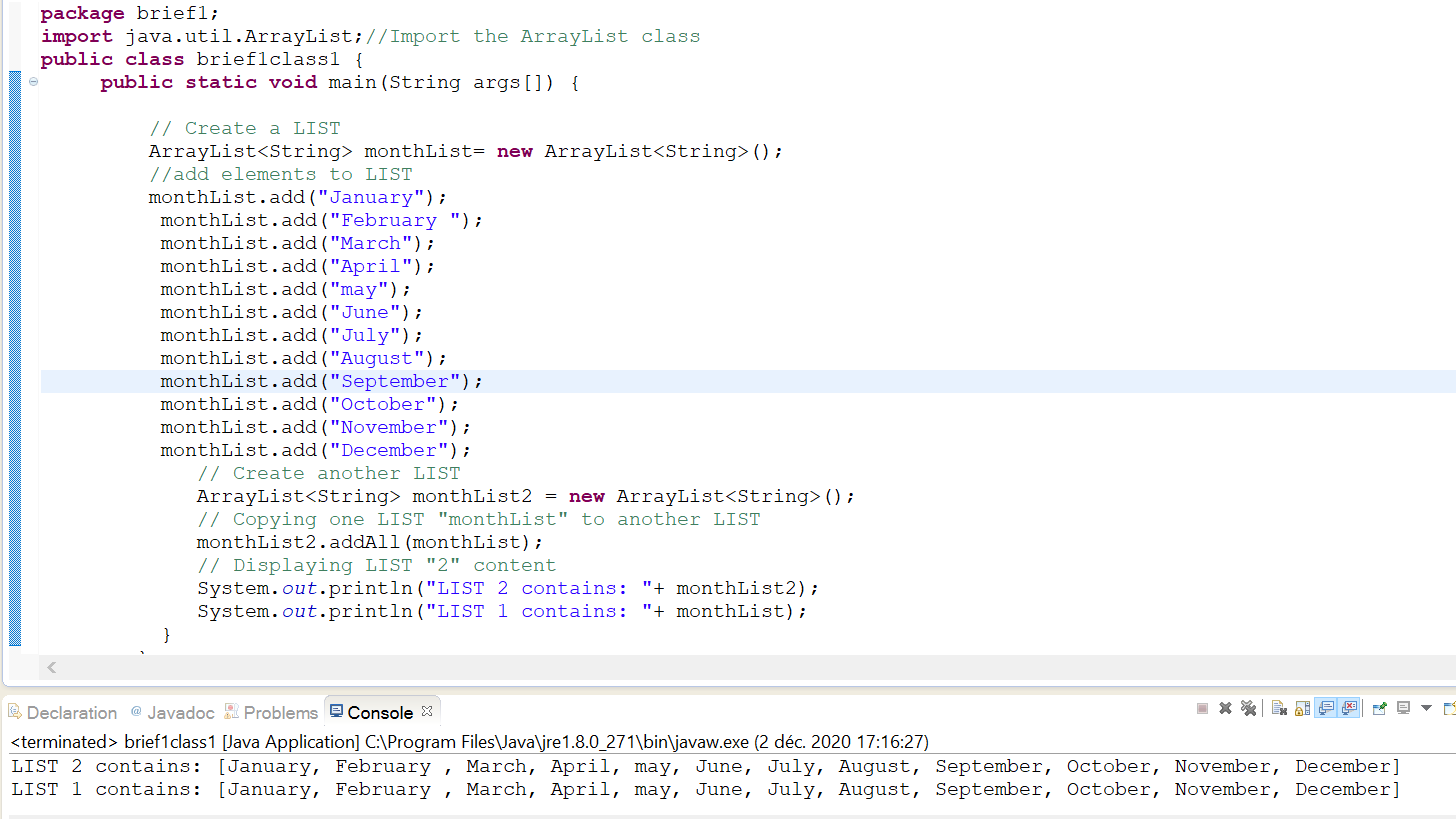
1. Rechercher un élément dans cette liste



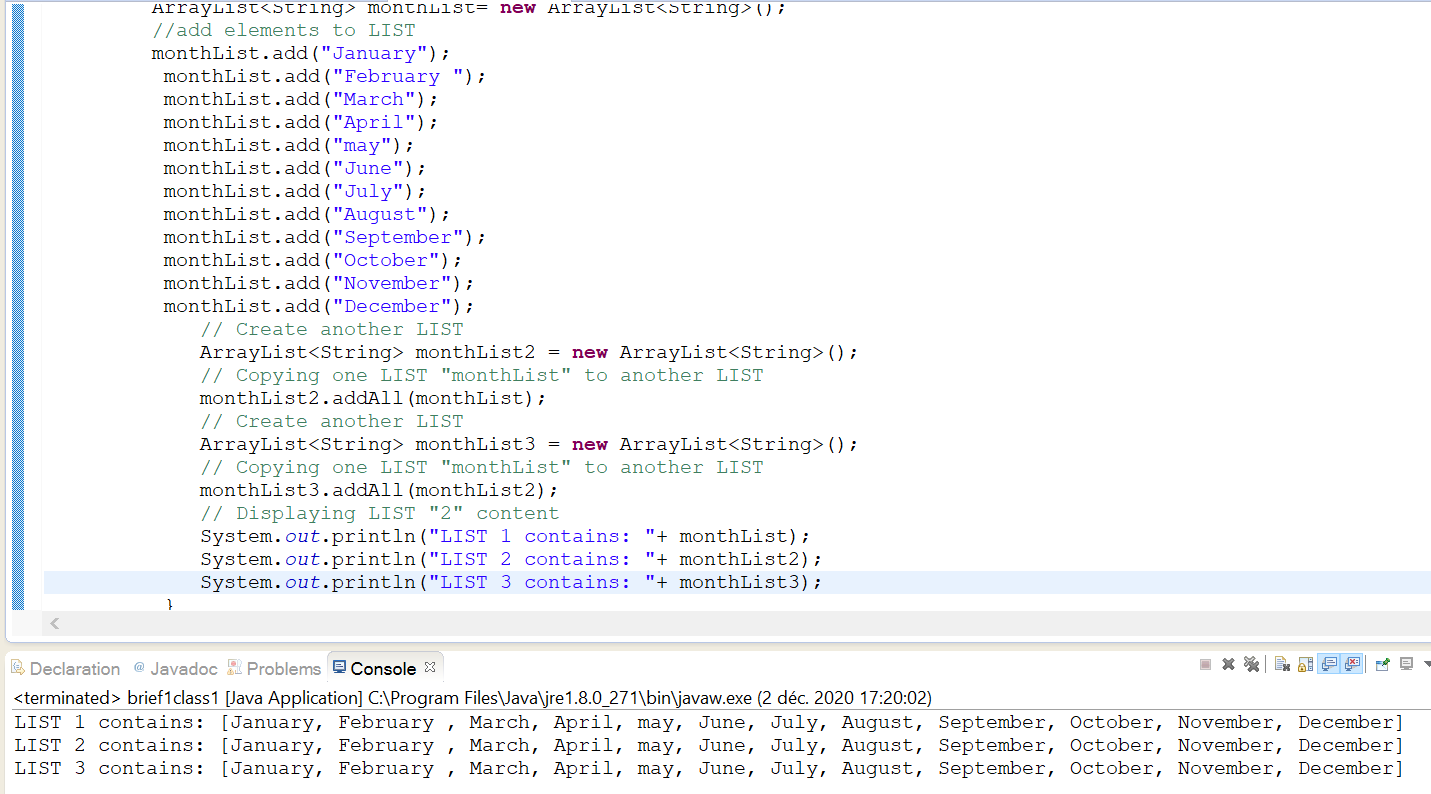
1. Trier cette liste



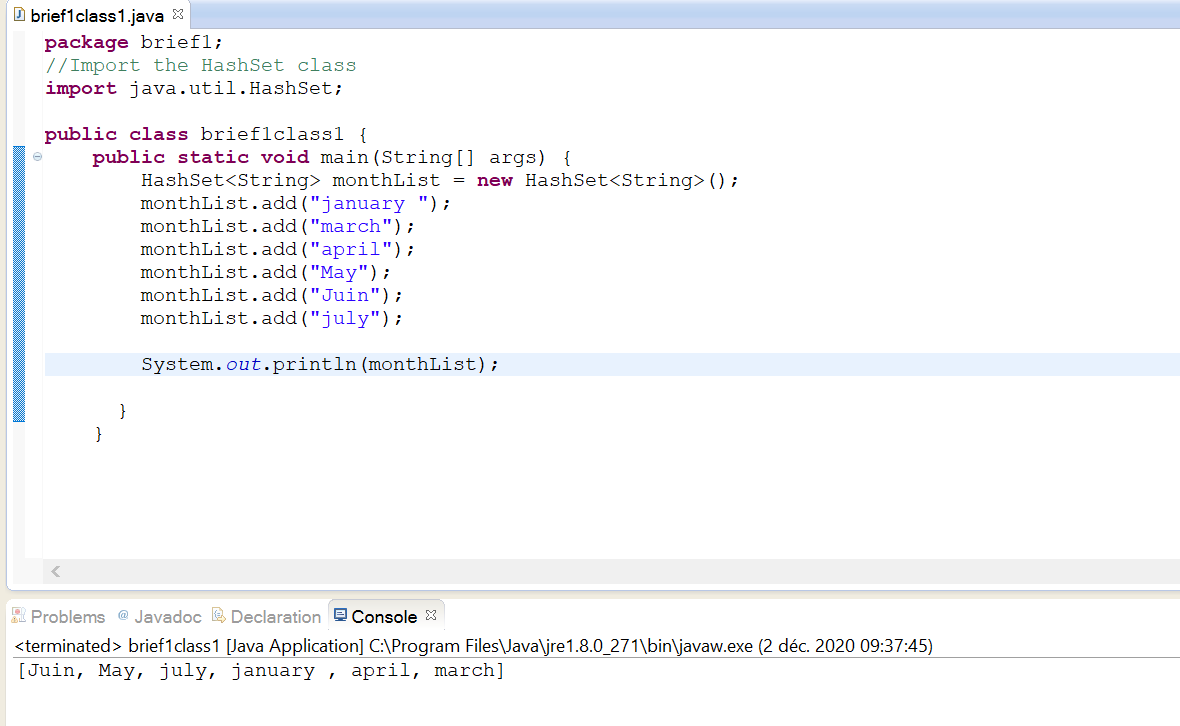
1. Copier cette liste dans une autre



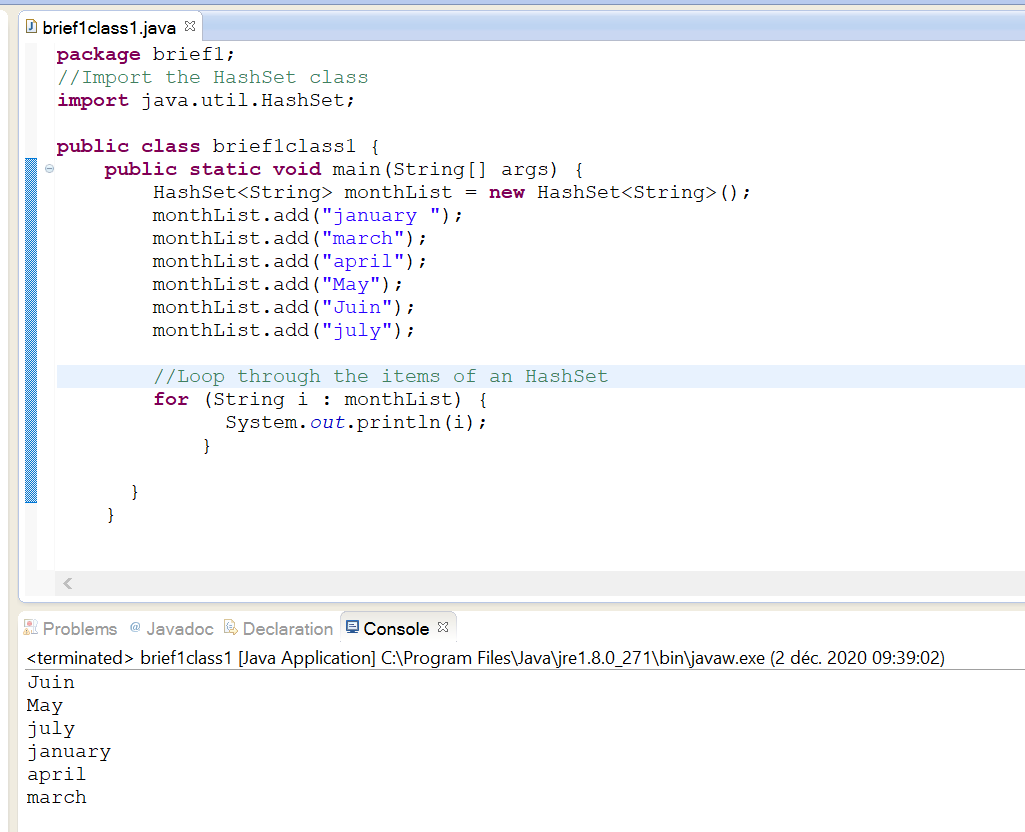
1. Cloner la liste dans une autre liste
2. Tester cette liste est vide ou non :



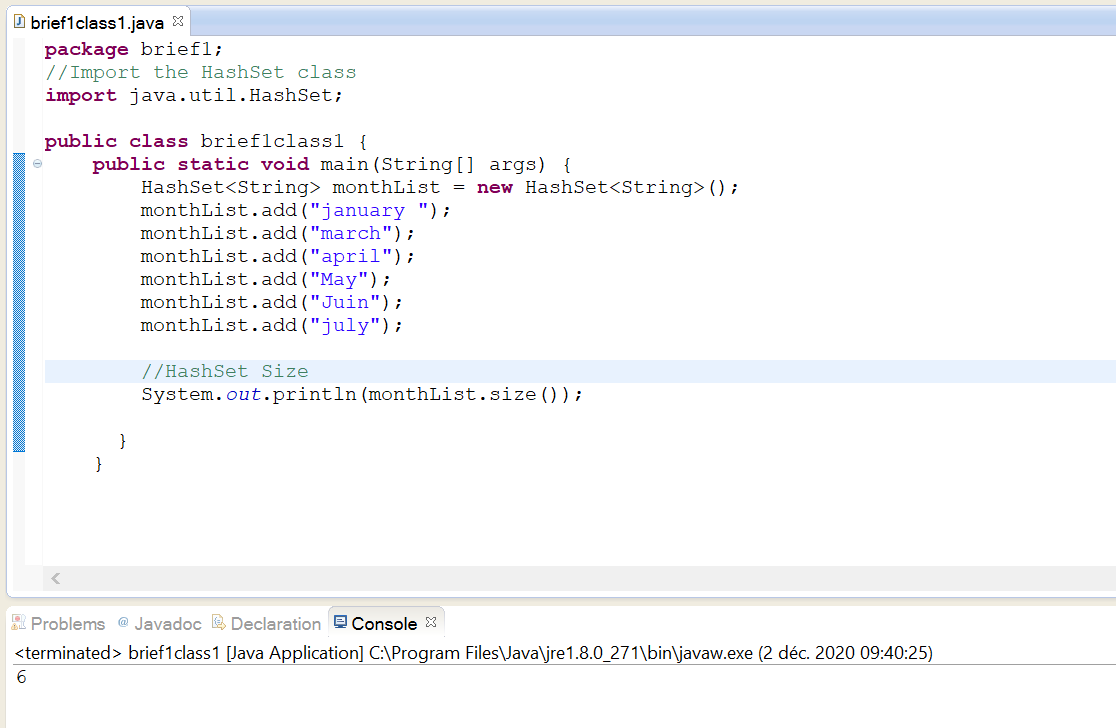
1. Créer une HashSet de mois, ajouter le nom des douze mois de l'année et afficher HashSet



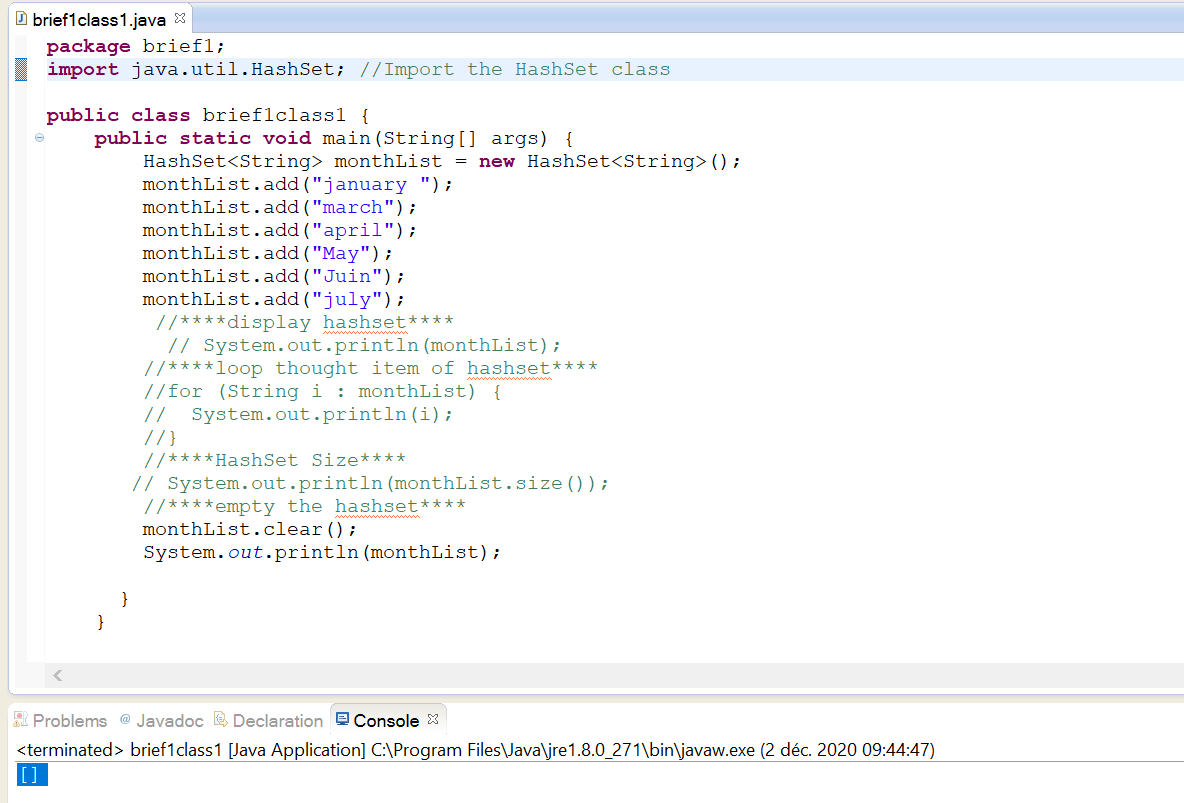
1. Parcourir tous les éléments de HashSet



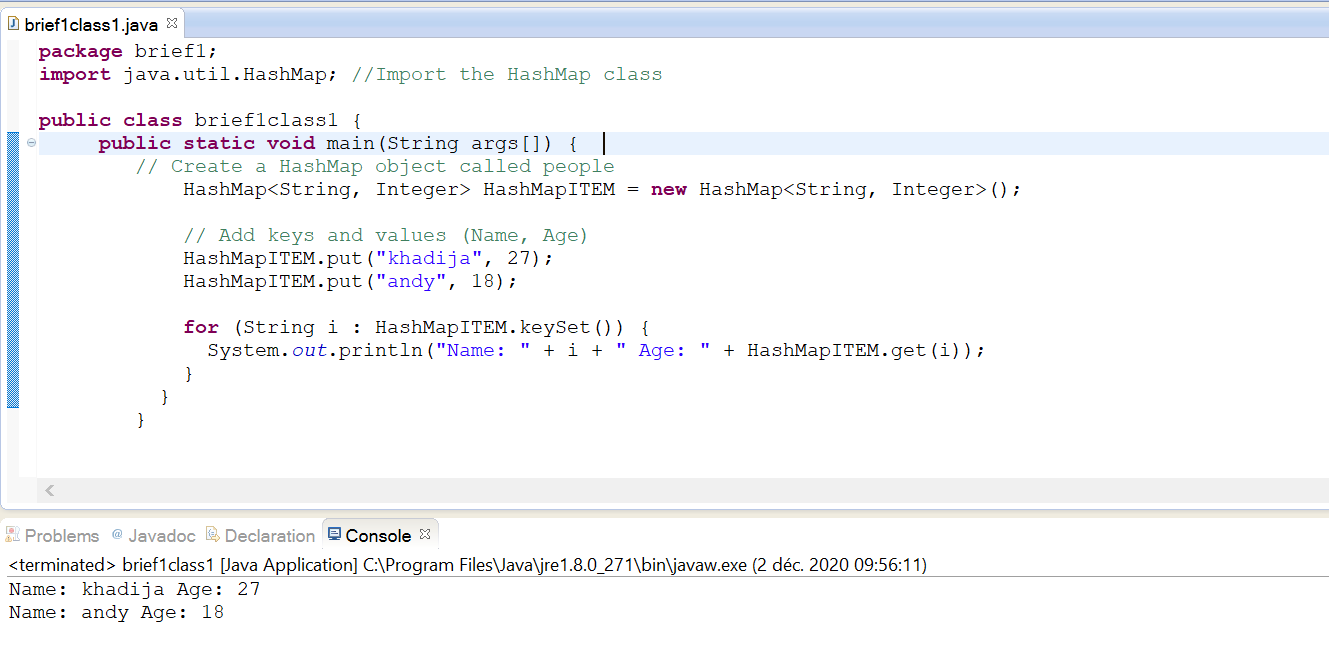
1. Obtenir le nombre d'éléments dans un ensemble de HashSet



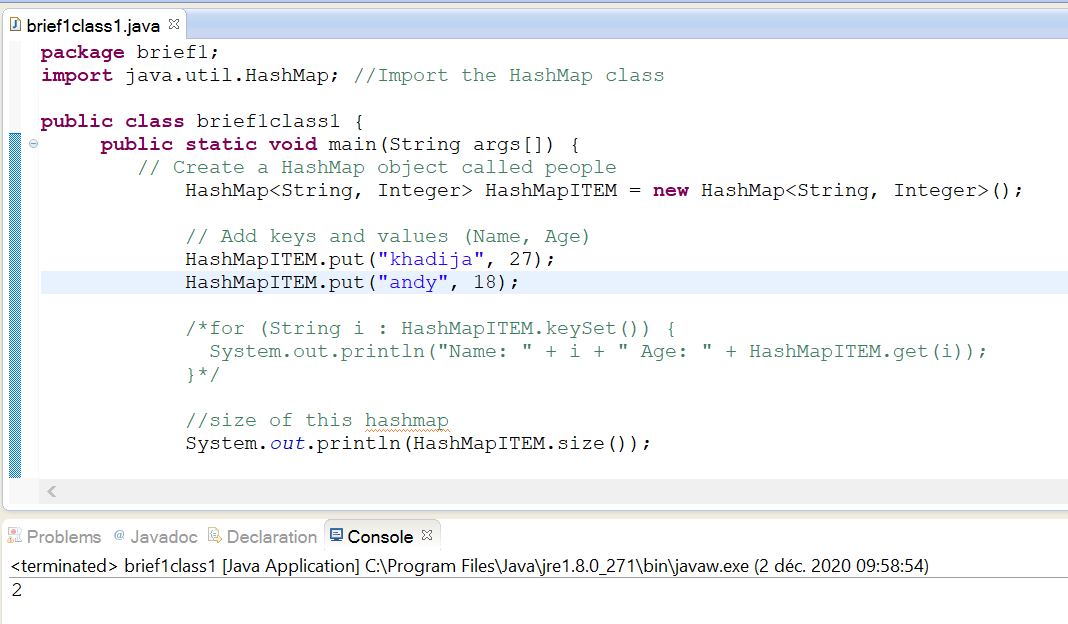
1. Vider cette HashSet
2. Tester que cette HashSet est vide ou non



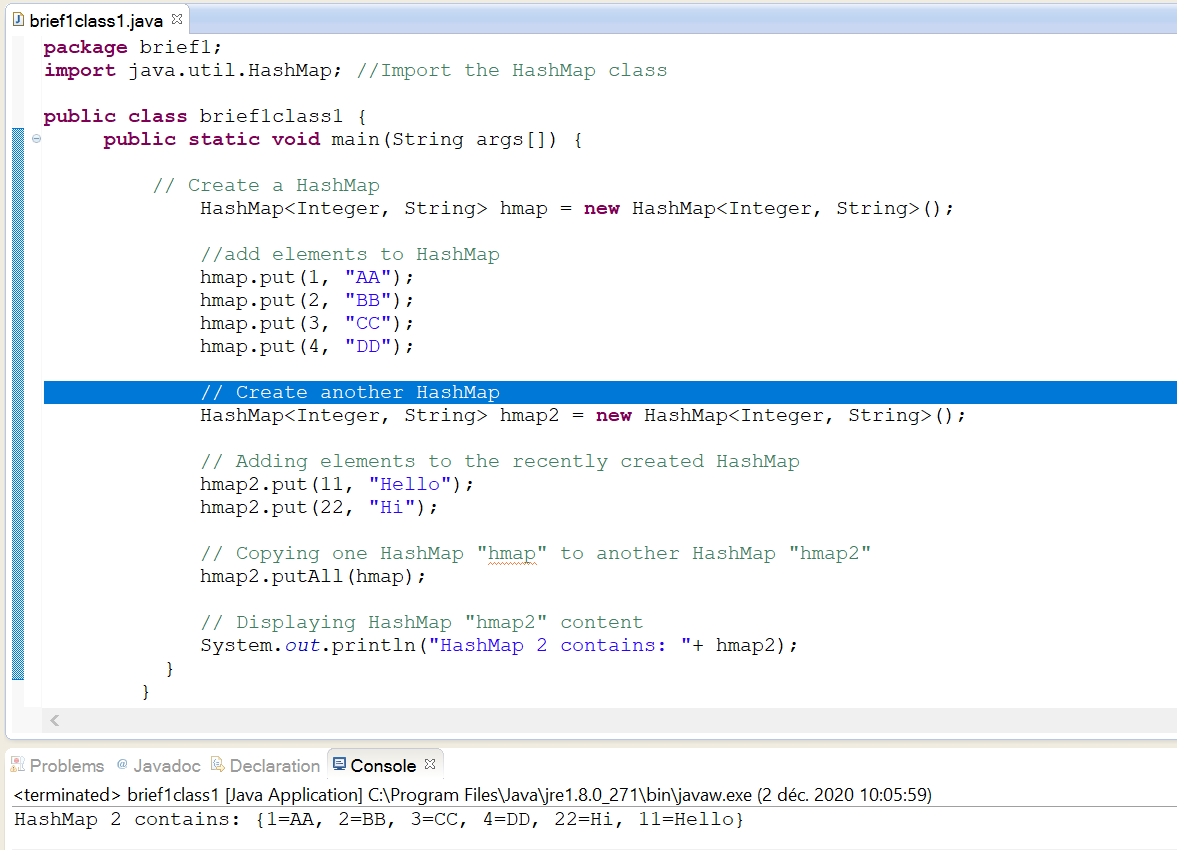
1. Écrivez un programme Java pour associer une valeur spécifiée à une clé spécifiée dans une HashMap.



1. Donner la taille de cette HashMap



1. Copier tous les éléments de cette HashMap dans une autre



1. Supprimer un élément spécifié de cette HashMap

