

Bouthayna Dhifaoui (bouthayna.dhfaoui@agil-it.fr)

INTERNE

Campagne : Java, Angular 2+ - Senior

Domaine(s) : Angular 2+, Java

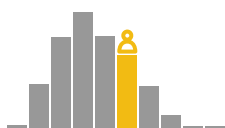
Langage : Français

Date : 15/02/2022

MEILLEUR QUE

75%

des professionnels



RANG

4 / 10



DURÉE

1h13 / 1h33



SCORE

880 / 1 720 (51%)

Angular 2+ 460 / 750pts (61%)

MEILLEUR QUE **98%** des professionnels

Connaissance du langage



460 / 750pts

Java 420 / 970pts (43%)

MEILLEUR QUE **47%** des professionnels

Connaissance du langage



287 / 602pts

Fiabilité



18 / 58pts

Résolution de problèmes



115 / 310pts

Question 1: String.isBlank()



Java



00:54 / 01:00



0 / 20 pts

? Question

Quel est le nom de la méthode de la classe `java.lang.String` qui permet de tester si une chaîne de caractères est vide ou ne contient que des caractères d'espacement?

```
String str1 = ""; // yes
String str1 = " "; // yes
String str1 = "Hello world."; // no
```

✎ Réponse

isEmpty

> Résultat



Réponse correcte

Connaissance du langage +20pts

✓ Réponse(s) correcte(s)

- `String.isBlank()`
- `.*isBlank.*`

Question 2: java.util.regex.Pattern



Java



00:26 / 00:45



0 / 20 pts

? Question

En Java, quelle classe permet de tester la validité d'une chaîne de caractères vis-à-vis d'une expression régulière telle que celle-ci :

```
String regex = "z.*t?";
```

📝 Réponse

- ☐ `java.util.regex.Pattern`
- ☐ `java.util.regex.Regex`
- ☒ `java.util.regex.Match`
- ☐ `java.util.regex.RegExp`

> Résultat



Réponse incorrecte

Connaissance du langage +20pts

Question 3: Attributs multiples



Java



01:18 / 01:30



0 / 40 pts



Question

Sélectionnez le code ci-contre qui permet à la classe ci-dessous de compiler sans erreur :

```
@Message(message = "Message 2")
@Message(message = "Message 1")
public class Example
{

}
```



Réponse



```
@Repeatable(Messages.class)
public @interface Message {
    public String message();
}
```

et

```
public @interface Messages {
    public Message[] value();
}
```



```
public @interface Message {
    public String message();
}
```



```
@Repeatable
public @interface Message {
    public String message();
}
```



```
@Repeatable(Messages.class)
public @interface Message {
    public String message();
}
```

et

```
public @interface Messages {
    public Message[] messages();
}
```



```
@Repeatable(Messages.class)
public attribute Message {
    public String message();
}
```

et

```
public attribute Messages {
    public Message[] value();
}
```



```
public attribute Message {
    public String message();
}
```



Résultat



Réponse incorrecte

Connaissance du langage ~~+40pts~~

Question 4: Héritage multiple d'interfaces



Java



00:28 / 00:30



40 / 40 pts



Question

```
public interface A extends B, C, D {}
```

Cette interface est correcte si B, C et D sont également des interfaces.



Réponse



Vrai



Faux



Résultat



Réponse correcte

Connaissance du langage +40pts

Question 5: Optimisation avec table de hachage



Java



01:45 / 02:30



40 / 40 pts



Question

Parmi les options suivantes, quelle classe utiliseriez-vous si vous deviez reprendre `GestionUtilisateurs` ?


```
class GestionUtilisateurs {  
  
    // identifiants : stocke les identifiants d'utilisateurs  
    ArrayList<Integer> identifiants = new ArrayList<Integer>();  
    // noms : stocke les noms d'utilisateurs  
    ArrayList<String> noms = new ArrayList<String>();  
  
    void ajouter(int identifiant, String nom) {  
        identifiants.add(identifiant);  
        noms.add(nom);  
    }  
  
    String trouverNom(int identifiant) {  
        for (int i = identifiants.size() - 1; i >= 0; i--) {  
            if (identifiants.get(i) == identifiant) {  
                return noms.get(i);  
            }  
        }  
        return null;  
    }  
}
```






Réponse

- ☐ Stack
- ☐ Vector
- ☐ ArrayList
- ☐ LinkedList
- ☒ HashMap
- ☐ HashSet

Résultat

 Réponse correcte
Connaissance du langage +40pts

Question 6: String buffers

 Java  00:27 / 00:30  0 / 60 pts


Question

Saisissez le nom d'une classe appartenant au package java.lang qui permet de concaténer efficacement les chaînes de caractères.

Réponse

Concat

Résultat

 Réponse correcte
Connaissance du langage +60pts

Réponse(s) correcte(s)

- java.lang.StringBuilder
- java.lang.StringBuffer
- (?i).*StringBuilder.*
- (?i).*StringBuffer.*

Question 7: Classification de paquets robotisée



Java



11:57 / 15:00



115 / 150 pts



Question

Objectif

Utiliser le bras robotique de l'usine pour trier les colis.

Règles

Vous travaillez pour une usine autonome. Votre objectif est de trier les colis qui arrivent sur le bon tas en fonction de leur volume et poids. Un colis est encombrant si son volume (Largeur x Hauteur x Profondeur) est supérieur ou égal à 1 000 000 cm³ ou si l'une de ses dimensions est supérieure ou égale à 150 cm. Un colis est lourd si sa masse est supérieure ou égale à 20kg. Vous devez répartir les colis dans 3 tas : STANDARD : les colis normaux (ni encombrants, ni lourds) seront traités normalement. SPECIAL : les colis lourds ou encombrants ne pourront pas être traités automatiquement. REJECTED : les colis à la fois encombrants et lourds seront refusés.

Implémentation

Implémentez la méthode `solve(width, height, length, mass)` (les unités sont le centimètre pour les dimensions et le kilogramme pour la masse). La méthode doit retourner une chaîne de caractères : le nom du tas où placer la boîte.

Conditions de Victoire

Les colis sont répartis sur les bons tas.

Conditions de Défaite

Votre programme indique une action invalide ou fausse.

Contraintes

$20 \leq \text{width, height, length} \leq 200$

$10 \leq \text{mass} \leq 1000$



Réponse

```
1 import java.util.*;
2 import java.io.*;
3 import java.math.*;
4
5 class Player {
6
7     public static String solve(int width, int height, int length, int mass) {
8         // Write your code here
9         // To debug: System.err.println("Debug messages...");
10        if(!estEncombrant(width,height,length) && !estMasse(mass))
11            return "STANDARD";
12        else if(estEncombrant(width,height,length) && estMasse(mass))
13            return "REJECTED";
14        else return "SPECIAL";
15    }
16    public static boolean estEncombrant(int width, int height, int length){
17
18        if(width*height*length >1000000 || width >=150 || height>=150 || length>=150)
19            return true;
20        return false;
21    }
22    public static boolean estMasse(int masse){
23
24        if(masse>=20)
25            return true;
26        else return false;
27    }
28    /* Ignore and do not change the code below */
29    // #region main
30    public static void main(String args[]) {
31        Scanner in = new Scanner(System.in);
32
33        // game loop
34        while (true) {
35            int width = in.nextInt();
36            int height = in.nextInt();
37            int length = in.nextInt();
38            int mass = in.nextInt();
39            PrintStream outputStream = System.out;
40            System.setOut(System.err);
41            String action = solve(width, height, length, mass);
42            System.setOut(outputStream);
43            System.out.println(action);
44        }
45    }
46    // #endregion
47 }
```

Résultat

-  Nombreuses boîtes
Résolution de problèmes +40pts
-  Seulement des boîtes STANDARD et REJECTED
Résolution de problèmes +40pts
-  Seulement des boîtes STANDARD et SPECIAL
Résolution de problèmes +35pts
-  Test cas limites
Résolution de problèmes ~~+35pts~~

Question 8: Utilisation d'un buffer/join à la place de '+'



Java



01:46 / 05:00



25 / 100 pts

? Question

`StringUtils.concat(String[] strings)` sert à joindre des chaînes de caractères bout à bout.

Par exemple, à partir d'un tableau contenant `"f", "o", "o", "bar"`, `concat` devrait retourner `"foobar"`.

Données : `strings` contient toujours au moins un élément.

Implémentez `StringUtils.concat(String[] strings)`.

📝 Réponse

```
1 class StringUtils {
2
3     /**
4      * Concatenates the given array of strings.
5      */
6     static String concat(String[] strings) {
7         String result="";
8         for(int i=0;i<strings.length;i++){
9             result+=strings[i];
10        }
11        return result;
12    }
13
14 }
```

> Résultat



Les chaînes sont concaténées

Connaissance du langage +25pts



Utilisation d'un buffer ou join() plutôt que '+'

Connaissance du langage -58pts



Utilisation de StringBuilder ou String.join() plutôt que StringBuffer

Connaissance du langage -17pts

Question 9: Intervalles



Java



12:00 / 12:00



0 / 200 pts

⚠ Le temps alloué à cette question s'est écoulé. La réponse du candidat a été automatiquement récupérée à la fin du décompte.



Question

Implémentez la méthode `findSmallestInterval(numbers)` qui retourne le plus petit intervalle positif entre deux éléments du tableau `numbers`.

Par exemple, si on considère le tableau `[1 6 4 8 2]`, le plus petit intervalle est `1` (différence entre `2` et `1`)

Contraintes:


`numbers` contient au moins deux éléments et au maximum 100 000 éléments. La solution qui privilégie la vitesse d'exécution pour les tableaux de grande taille obtiendra le plus de points. La différence entre deux éléments ne dépassera jamais la capacité d'un entier pour votre langage.





Réponse


```
1 import java.util.*;
2 import java.util.stream.Collectors;
3 import java.io.*;
4 import java.math.*;
5
6 class Solution {
7
8     public static int findSmallestInterval(int[] numbers) {
9         // Write your code here
10        // To debug: System.err.println("Debug messages...");
11        Set<Integer> set= new HashSet<>();
12        for(int i=0;i<numbers.length;i++){
13            set.add(Integer.valueOf(numbers[i]));
14        }
15        set= set.stream().sorted().collect(Collectors.toSet());
16        int smallInterval=0;
17        for(int i=0;i<set.size()-1;i++){
18            int interval=set[i+1]-set[i];
19            if(interval<smallInterval)
20                smallInterval=interval;
21        }
22
23
24        return smallInterval;
25    }
26
27    /* Ignore and do not change the code below */
28    // #region main
29    public static void main(String args[]) {
30        Scanner in = new Scanner(System.in);
31        int n = in.nextInt();
32        int[] numbers = new int[n];
33        for (int i = 0; i < n; i++) {
34            numbers[i] = in.nextInt();
35        }
36        PrintStream outputStream = System.out;
37        System.setOut(System.err);
38        int res = findSmallestInterval(numbers);
39        System.setOut(outputStream);
40        System.out.println(res);
41    }
42    // #endregion
43 }
```

Résultat

 Valideur simple
Résolution de problèmes ~~+40pts~~

 Couple inversé
Résolution de problèmes ~~+40pts~~

 Nombres négatifs
Résolution de problèmes ~~+40pts~~

 Très grand tableau
Résolution de problèmes ~~+40pts~~

 Grande différence
Fiabilité ~~+40pts~~

Question 10: Hello World



Java



01:55 / 05:00



200 / 200 pts



Question

```
console> java Echo Hello you !
```

```
Hello
```

```
you
```

```
!
```

```
console>
```

Écrivez le programme `Echo`.



Réponse

```
1 // Java code below
2 class Echo {
3     public static void main (String[] input){
4         for(int i=0;i<input.length;i++)
5             System.out.print(input[i]+"\\n");
6     }
7 }
8 }
```



Résultat



Le résultat affiché dans la console correspond

Connaissance du langage +182pts



Le résultat affiché prend en compte les arguments de la ligne de commande

Fiabilité +18pts

Question 11: Synchronisation d'un compteur



Java



02:00 / 02:00



0 / 100 pts

? Question

Transformez la méthode `Counter.increment()` pour qu'elle supporte l'accès concurrentiel de plusieurs threads.

✎ Réponse

```
1 class Counter {
2
3     private static int count = 0;
4
5     /**
6      * Increments count in a thread-safe manner.
7      */
8     public static int increment() {
9         try{
10             Thread thread= new Thread();
11             thread.
12         }catch(Exception e){
13
14         }
15         count = count + 1;
16         return count;
17     }
18 }
19 }
```

> Résultat



Accès concurrentiel synchronisé
Connaissance du langage +100pts

Question 12: Nouvelle application Angular



Angular 2+



00:15 / 00:30



20 / 20 pts



Question

Quelle ligne de commande permet de créer une nouvelle application Angular ?



Réponse



ng new appName



ng start appName



npm start appName



npm create appName



Résultat



Réponse correcte

Connaissance du langage +20pts

Question 13: Import de HttpClientModule



Angular 2+



00:14 / 00:30



20 / 20 pts



Question

Quel module doit-on importer pour utiliser le service `HttpClient` ?



Réponse



`HttpClientModule`



`HttpModule`



`RequestModule`



`TransportModule`



`ClientModule`



Résultat



Réponse correcte

Connaissance du langage +20pts

Question 14: Trouver le nom



Angular 2+



00:23 / 01:00



20 / 20 pts



Question

Quel(s) nom(s) seront affichés par ce code ?


```
1 import { Component } from '@angular/core';
2
3 @Component({
4   selector: 'my-app',
5   template: `
6     <div *ngFor="let name of names">
7       <div *ngIf="name.length > 6">
8         {{name}}
9       </div>
10    </div>
11  `
12 })
13 export class AppComponent {
14   names = ['Theresa', 'Morris', 'Duane', 'Taz'];
15 }
```



Réponse

- ☒ Theresa
- ☐ Morris
- ☐ Duane
- ☐ Theresa, Morris, Duane
- ☐ Taz
- ☐ Nothing will be printed
- ☐ Morris, Duane

Résultat

 Réponse correcte
Connaissance du langage +20pts

Question 15: Interpolation



Angular 2+



00:18 / 00:50



20 / 20 pts



Question

Quelle concept Angular est mis en oeuvre dans le template ci-dessous à la ligne 6 pour afficher la propriété `name` ?

```
1 import { Component } from '@angular/core';
2
3 @Component({
4   selector: 'my-app',
5   template: `
6     <h1>Hello {{name}}</h1>
7   `
8 })
9 export class AppComponent {
10   name = 'John Doe';
11 }
```



Réponse

- ☒ string interpolation
- ☐ template string
- ☐ template literal
- ☐ event binding



Résultat



Réponse correcte

Connaissance du langage +20pts

Question 16: Directive ngIf



Angular 2+



00:20 / 01:00



20 / 20 pts



Question

Qu'arrivera-t-il si on clique plusieurs fois sur le bouton tel que défini à la ligne 6 ?

```
1 import { Component } from '@angular/core';
2
3 @Component({
4   selector: 'my-app',
5   template: `
6     <button type="button" (click)="onClick()">Click Me</button>
7     <h1 *ngIf="show">
8       {{name}}
9     </h1>
10  `
11 })
12 export class AppComponent {
13   name = 'John Doe';
14   show = true;
15
16   onClick() {
17     this.show = !this.show;
18   }
19 }
```



Réponse

- ☐ La propriété name est affichée
- ☐ La propriété name est cachée
- ☒ La propriété name est affichée une fois sur deux
- ☐ La propriété name est effacée

> Résultat

✓ Réponse correcte
Connaissance du langage +20pts

Question 17: Pipe currency

Angular 2+ 00:30 / 00:50 40 / 40 pts

? Question

On suppose que la locale par défaut est `en-US`. Qu'affichera l'expression de la ligne n°6 ?

```
1 import { Component } from '@angular/core';
2
3 @Component({
4   selector: 'my-app',
5   template: `
6     {{ cost | currency }}
7   `
8 })
9 export class AppComponent {
10   cost = '200.13';
11 }
```

✎ Réponse

\$200.13

> Résultat

✓ Réponse correcte
Connaissance du langage +40pts

✓ Réponse(s) correcte(s)

- \$200.13

Question 18: Définition des routes filles



Angular 2+



00:30 / 00:30



0 / 40 pts

⚠ Le temps alloué à cette question s'est écoulé. La réponse du candidat a été automatiquement récupérée à la fin du décompte.



Question

Au sein de l'objet de type `Routes`, quelle propriété permet de définir des routes filles ?



Réponse

- ☐ loadChildren
- ☐ route
- ☐ children
- ☒ childRoutes



Résultat



Réponse incorrecte

Connaissance du langage ~~+40pts~~

Question 19: Écouter les événements de l'hôte



Angular 2+



00:15 / 00:45



40 / 40 pts



Question

Quel décorateur peut-on utiliser pour écouter les événements de l'élément hôte depuis une directive fille?



Réponse



@HostListener



@Listener



@Target



@Event



Résultat



Réponse correcte

Connaissance du langage +40pts

Question 20: [(ngModel)] import



Angular 2+



00:15 / 00:30



40 / 40 pts



Question

Quel module doit-on importer pour pouvoir utiliser [(ngModel)] ?



Réponse

- ☐ NgModule
- ☐ BrowserModule
- ☒ FormsModule
- ☐ InputModule



Résultat



Réponse correcte

Connaissance du langage +40pts

Question 21: Accéder aux paramètres de route



Angular 2+



00:25 / 00:30



0 / 40 pts



Question

Quel service peut-on utiliser pour récupérer les paramètres de route depuis un composant ?



Réponse



Route



Router



ActivatedRoute



CurrentRoute



Résultat



Réponse incorrecte

Connaissance du langage +40pts

Question 22: @Input et événements @Output



Angular 2+



19:14 / 20:00



2x (3 sec)



0 / 150 pts

? Question

On vous demande de créer un composant Angular simple nommé `CounterComponent` ayant `counter-component` pour sélecteur.

L'objectif de ce composant est de :

Prendre un message en "input" Lire une valeur d'intervalle saisie par l'utilisateur (entier, en millisecondes) Envoyer le message en tant qu'événements de manière périodique vers son "output"

Au niveau HTML, le composant est composé d'un champ `<input>` avec `id="intervalInput"` et d'un bouton `<button>` avec `id="intervalSetButton"`. L'utilisateur peut modifier l'intervalle en saisissant une valeur dans le champ puis en cliquant sur le bouton.

Le composant doit avoir un `@Input` nommé `message` et un `@Output` nommé `tick`. Il doit générer des événements de type "string" vers `tick` à intervalle régulier correspondant à la valeur saisie par l'utilisateur. La chaîne de caractère utilisée pour générer les événements provient de l'"input" `message`.


Le composant doit être capable de gérer les changements du message et les changements de la valeur de l'intervalle. Notes : Vous devez modifier le template ainsi que la classe du composant. Un bloc de "Preview" est disponible pour vous permettre de déboguer votre code. Il affiche votre composant ci-dessous. Vous pouvez l'ouvrir et le modifier à votre guise. Pour vous aider, ce bloc contient un composant parent qui manipule le composant que vous devez créer (affichage d'un compteur à côté du message) Ce bloc de "Preview" n'est pas pris en compte dans le calcul du score de votre code.



Réponse

```
1 // Angular 8.x code
2 import { Component, NgModule, Output, Input, EventEmitter } from '@angular/core';
3 //import { NgForms } from '@angular/common';
4
5
6
7 @Component({
8   selector: 'counter-component',
9   template: `
10     <input id="intervalInput"/>
11     <button id="intervalSetButton" (click)="counterTick(message)">Set interval</button>
12     <div></div>
13   `
14 })
15 export class CounterComponent {
16   @Input() message: string;
17   @Output() tick=new EventEmitter<string>();
18   public counterTick(message: string): void {
19     this.tick.emit(message);
20   }
21 }
22
23
24 // #region Preview
25 @Component({
26   selector: 'display-component',
27   template: `
28     <counter-component [(NgModule)]="message" (tick)=counterTick($event)></counter-
29     component>
30     <div>{{message}} {{counter}}</div>
31   `
32 })
33 export class DisplayComponent {
34   public counter: number = 0;
35   public message: string='Hello world';
36
37   public counterTick(message: string): void {
38     this.message = message;
39     this.counter++;
40     //this.tick.emit(message);
41   }
42 }
43
44 @Component({
45   template: `<display-component></display-component>`
46 })
47 export class PreviewComponent { }
48 // #endregion Preview
49
50 // #region Module declaration - Do not Change
51 @NgModule({
52   declarations: [PreviewComponent, DisplayComponent, CounterComponent],
53   entryComponents: [PreviewComponent]
54 })
55 export class PreviewModule { }
56 // #endregion Module declaration
```

Résultat

 Compile
Connaissance du langage +0pts

 Prend en compte les changements d'input
Connaissance du langage ~~+30pts~~

 Prend en compte les changements d'intervalle
Connaissance du langage ~~+30pts~~

 Envoie les événements toutes les 10ms
Connaissance du langage ~~+90pts~~

Question 23: Utilisation des pipes (percent, currency, date)



Angular 2+



14:55 / 20:00



240 / 300 pts

? Question

On vous demande de terminer le composant `TransactionDetailsComponent`. Vous devez uniquement modifier la partie template du composant.

L'objectif de ce composant est d'afficher la date, le montant, la devise et les frais associés à une transaction, chacun dans un format spécifique.

Vous devez afficher 3 divs :

Le div "Fee" (frais)

L'id de ce div doit être `fee`. Il affiche les frais `fee` associés à la transaction sous forme de pourcentage : S'il y a moins de 2 chiffres pour la partie entière de la valeur en pourcentage, vous devez combler avec des zéros à gauche. S'il y a moins de 2 chiffres dans la partie décimale, vous devez combler avec zéros à droite. S'il y a plus de 3 chiffres dans la partie décimale, vous devez arrondir à 3 chiffres. On utilise un point `.` en tant que séparateur décimal.

Par exemple `0.031234` s'affiche `03.123%`.

Le div "Amount" (montant)

L'id de ce div doit être `amount`. Il affiche les frais `amount` et la devise `currency` de la transaction. Le symbole de la devise associée au code `currency` (par exemple `€` pour `EUR`) est affiché avant les frais. Les frais sont formatés comme suit :

S'il y a moins de 9 chiffres pour la partie entière, vous devez combler avec des zéros à gauche. Le séparateur des milliers doit être une virgule `,`. S'il y a moins de 2 chiffres dans la partie décimale, vous devez combler avec zéros à droite. S'il y a plus de 2 chiffres dans la partie décimale, vous devez arrondir à 2 chiffres.

Par exemple `currency=EUR`, `amount=312.562` s'affiche `€000,000,312.56`.

Le div "Time" (date)

L'id de ce div doit être `time`. La date et l'heure de la transaction doivent être affichés dans ce format inhabituel : 'ww: yyyy MMMMM dd hh-mm-ss'

Notes : Un bloc de "Preview" est disponible pour vous permettre de déboguer votre code. Il affiche votre composant ci-dessous. Vous pouvez ouvrir ce bloc et le modifier à votre guise. Le bloc de "Preview" n'est pas pris en compte dans le calcul du score de votre code.



Réponse

```
1 // Angular 8.x code
2 import { Component, Input, NgModule } from '@angular/core';
3 import { CommonModule } from '@angular/common';
4
5 @Component({
6   selector: 'transaction-component',
7   template: `
8     <div id="fee">{{fee| percent:'2.2-3'}}</div>
9     <div id="amount">{{amount|currency:'EUR':'symbol':'9.2-3'}}</div>
10    <div id="time">{{timeOfTransaction|date:'ww: yyyy MMMMM dd hh-mm-ss'}}</div>
11  `
12 })
13
14 export class TransactionDetailsComponent {
15
16   @Input()
17   public currency: string;
18
19   @Input()
20   public timeOfTransaction: Date;
21
22   @Input()
23   public amount: number;
24
25   @Input()
26   public fee: number;
27
28 }
29
30 // #region Preview
31 @Component({
32   template: `<transaction-component [fee]=0.02 [amount]=123.45 [currency]=" 'EUR' "
33   [timeOfTransaction]='getPresetDate()'></transaction-component>`
34 })
35 export class PreviewComponent {
36   public getPresetDate(){
37     return new Date(1997,6,1,12,32);
38   }
39 }
40 // #endregion Preview
41
42 // #region Module declaration - Do not Change
43 @NgModule({
44   imports: [CommonModule],
45   declarations: [PreviewComponent, TransactionDetailsComponent],
46   entryComponents: [PreviewComponent]
47 })
48 export class PreviewModule { }
49 // #endregion Module declaration
```

Résultat

- ✓ Compile
Connaissance du langage +0pts
- ✓ Affichage pourcentage
Connaissance du langage +60pts
- ✓ Affichage de date
Connaissance du langage +60pts
- ✓ Affichage EUR
Connaissance du langage +60pts
- ✗ Affichage JPY
Connaissance du langage ~~+60pts~~
- ✓ 3 décimales pour le pourcentage
Connaissance du langage +60pts

Glossaire

Connaissance du langage

La mesure de cette compétence permet de déterminer l'expérience du candidat dans la pratique d'un langage de programmation. **Privilégiez cette compétence si, par exemple, vous recherchez un développeur qui devra être rapidement opérationnel.**

Modélisation

Cette mesure fournit une indication sur la capacité du candidat à appliquer des solutions standard pour résoudre des problèmes récurrents. Un développeur ayant un bon niveau dans cette compétence augmentera la qualité (maintenabilité, évolutivité) de vos applications. Cette compétence ne dépend pas spécifiquement d'une technologie. **Privilégiez cette compétence si, par exemple, vous recherchez un développeur qui sera amené à travailler sur les briques qui structurent vos applications, à anticiper les besoins de demain pour développer des solutions pérennes.**

Résolution de problèmes

Cette compétence correspond aux aptitudes du candidat à comprendre et à structurer son raisonnement pour trouver des solutions à des problèmes complexes. Cette compétence ne dépend pas spécifiquement d'une technologie. **Privilégiez cette compétence si, par exemple, vos applications ont une composante technique importante (R&D, innovation).**

Fiabilité

La fiabilité caractérise la capacité du candidat à réaliser des solutions qui prennent en compte les cas particuliers. Plus cette compétence est élevée, plus vos applications sont robustes (moins de bugs).