

Java Developer Take Home Quiz

Given:

A csv file (eng-climate-summary.csv) with historic weather-related information.

Requirement Details:

Create 2 web pages to display the following information (extracted from the provided csv):

1. Create a **table** with columns **Station_Name**, **Date** and **Mean_Temp**
 - a. Create a **filter** for Date from the above table. Whereby a user can filter results provided within a date range.
2. Create a **details page** (upon clicking a value in the **Mean_Temp** column for one of the Stations) to display **Station_Name**, **Province**, **Date**, **Mean_Temp**, **Highest_Monthly_Max_Temp** and **Lowest_Monthly_Min_Temp** of the selected station.

Implementation Details:

- Add test cases to validate your solution
- Add a README file and describe the steps to compile/test/run your solution
- Ensure that your solution runs from the command line
- Display a meaningful message if a user types a wrong URL to access a resource in your web application

Include the following technologies/frameworks in your solution:

Technology / Framework	Mandatory	Nice to Have	Version
Java	X		8
Spring Boot	X		
Spring, Spring MVC	X		5.0 (or higher)
Thymeleaf (or equivalent)	X		
JUnit/Mockito (or equivalent)	X		
Ant / Maven / Gradle (to build/test/run your project)	X		
Logging framework		X	
CSS / JavaScript		X	
AJAX or JSON		X	
Client-Side and Server-Side Validation		X	

Submission Details:

- Please share the quiz results via GitHub and reply with location and access details.

Développeur Java – Test à domicile

Matériel fourni

Fichier csv (intitulé « eng-climate-summary.csv ») contenant des données météorologiques.

Exigences

Créer deux pages Web pour afficher les données suivantes (extraites du fichier csv fourni) :

3. Créer une **table** contenant les colonnes **Station_Name** (nom de la station), **Date** et **Mean_Temp** (température moyenne).
 - a. Créer un **filtre** par **Date** à partir de la table préparée au point précédent et permettant à l'utilisateur de filtrer les résultats fournis dans une plage de dates.
4. Créer une **page de détails** (accessible en cliquant sur une valeur de la colonne **Mean_Temp** [température moyenne] d'une des stations) pour afficher les valeurs **Station_Name** (nom de la station), **Province**, **Date**, **Mean_Temp** (température moyenne), **Highest_Monthly_Max_Temp** (température maximale la plus élevée d'un mois) et **Lowest_Monthly_Min_Temp** (température minimale la plus basse d'un mois) de la station sélectionnée.

Exécution

- Fournir des scénarios d'essai pour valider votre solution.
- Décrire les étapes à effectuer pour compiler, tester et exécuter votre solution dans un fichier « Lisez-moi ».
- Votre solution doit être exécutable à partir de la ligne de commande.
- Faire afficher un message pertinent dans le cas où un utilisateur inscrit une adresse URL erronée pour accéder à une ressource dans votre application Web.

Technologies et cadres à utiliser

Technologie ou cadre	Obligatoire	Atout	Version
Java	X		8
Spring Boot	X		
Spring, Spring MVC	X		5.0 (ou plus)
Thymeleaf (ou l'équivalent)	X		
JUnit/Mockito (ou l'équivalent)	X		
Ant/Maven/Gradle (pour créer/tester/exécuter votre projet)	X		
Logging framework		X	
CSS/JavaScript		X	
AJAX ou JSON		X	
Validation côté client et côté serveur		X	

Envoi

- Partagez les résultats de votre test à l'aide de GitHub et répondez en fournissant l'emplacement et les données d'accès.