



Problem Solving (DS4)

هذا البرنامج التدريبي مُصاغ بعناية لتمكين المتدربين من تطوير قدراتهم الفكرية على غرار المبرمجين المحترفين، والتعاون بكفاءة ضمن فريق محترف في شركة "جمال تك" أو أي مؤسسة متعددة الجنسيات أخرى. نظرًا لأهمية اللغة الإنجليزية في بيئة العمل العالمية، يتم تقديم المحتوى التدريبي بالإنجليزية. لا يشترط إتقان اللغة بشكل كامل، لكن من الضروري امتلاك القدرة الكافية لفهم المتطلبات وتنفيذها بشكل فعال. يُمكن للمتدربين استخدام مترجم جوجل أو الاستعانة بـ "شات جي بي تي" للتغلب على أية عقبات لغوية، المهم هو الفهم الدقيق للمطلوب وتحقيقه بنجاح.

لتعظيم الاستفادة من التدريب، يُنصح بمحاولة حل التمارين بشكل مستقل لمدة ساعة واحدة على الأقل قبل الرجوع إلى الحل المرفق في نهاية الملف.

قد يتضمن الحل كودًا برمجيًا غير مفسر بعد، والغرض من ذلك هو تشجيعك على محاولة فهم الأكواد البرمجية الجديدة التي لم تتعرض لها من قبل. هذه المهارة ضرورية في سوق العمل، حيث تتطور لغات البرمجة باستمرار ويظهر كل يوم لغات جديدة. ستواجه دائمًا أكوادًا لم تدرسها من قبل، ومن المهم أن تكون قادرًا على فهمها بنفسك دون الحاجة إلى دراسة مسبقة. يمكنك الاستعانة بمحرك البحث جوجل، أو استخدام ChatGPT، أو حتى اللجوء لأصدقائك للمساعدة. الهدف الأساسي هو أن تصل إلى فهم معنى كل كود بأي طريقة ممكنة لتتمكن من إيجاد موقعك في سوق العمل.

إن وجود كود برمجي غير مفسر يشكل تحديًا يتوجب عليك إيجاد حل له. هذا النوع من التدريبات يعد جزءًا أساسيًا من تدريبات 'Problem Solving'، التي تهدف إلى تمكينك من أداء عملك بفاعلية بغض النظر عن التحديات والعقبات. هذه القدرة على حل المشكلات هي ما يتمتع به العاملون في 'جمال تك'، ومن الضروري أن تطور في نفسك هذه المهارة لتصبح عضوًا فعالًا في فريق عمل 'جمال تك'.

Gammal Tech's Agile Journey

- Background:

Gammal Tech, a cutting-edge software development company, is revered for its innovative solutions and exceptional work environment. As a leader in the software industry, Gammal Tech is dedicated to implementing Agile methodologies in its projects. Your task is to assist Gammal Tech in automating part of their Agile process.

- Problem Statement:

You are to build a program for Gammal Tech that takes the current stage of any software project as input and prints the next stage in the software lifecycle, as per Agile methodology. The stages in Agile methodology are: "Requirement", "Design", "Development", "Testing", "Deployment", and "Maintenance". Once the "Maintenance" stage is reached, the next stage should loop back to "Requirement".



Constraints:

The input stage will always be a valid stage name in the Agile lifecycle.

The program should be case-insensitive with respect to input.

Input Format:

- A single line containing the name of the current stage (e.g., "Testing").

Output Format:

- A single line displaying the name of the next stage in the lifecycle (e.g., "Deployment").

Sample Input

```
Development
```

Sample Output

```
Testing
```

Explanation:

Given the current stage as "Development", the next stage in the Agile lifecycle is "Testing".

لتعظيم الاستفادة من التدريب، يُنصح بمحاولة حل التمرين بشكل مستقل لمدة ساعة واحدة على الأقل قبل الرجوع إلى الحل المرفق



C++ Solution:

```
#include <iostream>
#include <string>
#include <map>

using namespace std;

string getNextStage(const string& currentStage) {
    map<string, string> lifecycle = {
        {"Requirement", "Design"},
        {"Design", "Development"},
        {"Development", "Testing"},
        {"Testing", "Deployment"},
        {"Deployment", "Maintenance"},
        {"Maintenance", "Requirement"}
    };

    string lowerCaseStage = currentStage;
    for (char &c : lowerCaseStage) {
        c = tolower(c);
    }

    // Capitalizing the first letter for matching key in the map
    lowerCaseStage[0] = toupper(lowerCaseStage[0]);

    return lifecycle[lowerCaseStage];
}

int main() {
    string currentStage;
    getline(cin, currentStage);

    string nextStage = getNextStage(currentStage);
    cout << nextStage << endl;

    return 0;
}
```