
سلسلة الخوارزميات وحل المشاكل - المستوى الثاني
PROGRAMMING ADVICES
LEARN THE RIGHT WAY
26+ Years of Experience
Mohammed Abu-Hadhoud
MBA, PMOC, PgMP®, PMP®, PMI-RMP®, CM, ITILF, MCPD, MCSD
حقوق النشر محفوظة، أسعار الكورسات في المنصة هي أسعار رمزيه جدا، ارجو عدم نشر هذه الوثيقة لان نشرها سيمنعنا من الاستمرار في تقديم العلم للآخرين
ارجو عدم استخدام هذه الوثيقة من غير وجه حق لأنك ستحرم الاف الناس من التعلم
ProgrammingAdvices.com

```cpp
#include <iostream>
using namespace std;

enum enCharType { SamallLetter = 1, CapitalLetter = 2,
SpecialCharacter = 3, Digit = 4 };

int RandomNumber(int From, int To)
{
    //Function to generate a random number
    int randNum = rand() % (To - From + 1) + From;
    return randNum;
}

char GetRandomCharacter(enCharType CharType)
{
    switch (CharType)
    {

    case enCharType::SamallLetter:
    {
        return char(RandomNumber(97, 122));
        break;
    }
    case enCharType::CapitalLetter:
    {
        return char(RandomNumber(65, 90));
        break;
    }
    case enCharType::SpecialCharacter:
    {
        return char(RandomNumber(33, 47));
        break;
    }
    case enCharType::Digit:
    {
        return char(RandomNumber(48, 57));
        break;
    }

    }
}
```

```
void PrintStringArray(string arr[100], int arrLength)
{

    cout << "\nArray elements:\n\n";
    for (int i = 0; i < arrLength; i++)
    {
        cout << "Array[" << i << "] : ";
        cout << arr[i] << "\n";
    }
    cout << "\n";

}

string GenerateWord(enCharType CharType, short Length)

{
    string Word;

    for (int i = 1; i <= Length; i++)

    {

        Word = Word + GetRandomCharacter(CharType);

    }
    return Word;
}

string  GenerateKey()
{

    string Key = "";

    Key = GenerateWord(enCharType::CapitalLetter, 4) + "-";
    Key = Key + GenerateWord(enCharType::CapitalLetter, 4) + "-";
    Key = Key + GenerateWord(enCharType::CapitalLetter, 4) + "-";
    Key = Key + GenerateWord(enCharType::CapitalLetter, 4);


    return Key;
}
```

```
void FillArrayWithKeys(string arr[100], int arrLength)
{

    for (int i = 0; i < arrLength; i++)
        arr[i] = GenerateKey();

}



int ReadPositiveNumber(string Message)
{
    int Number = 0;
    do
    {
        cout << Message << endl;
        cin >> Number;
    } while (Number <= 0);
    return Number;
}


int main() {
    //Seeds the random number generator in C++, called only once
    srand((unsigned)time(NULL));

    string arr[100];
    int  arrLength = 0;

    arrLength = ReadPositiveNumber("How many keys do you want to
generate?\n");

    FillArrayWithKeys(arr, arrLength);

    PrintStringArray(arr, arrLength);

    return 0;
}
```