

Plateforme de Développement Continu

Comprendre la Théorie pour mieux Pratiquer
Sciences de l'Ingénieur
Cours - Tutoriels

PHP

Apprendre le PHP

J'aime, Je partage
Montez en Compétences

Auteur

Je suis **Gérard KESSE**,
Ingénieur en Développement Informatique C/C++/Qt,
Avec à la fois des compétences en Systèmes Embarqués et en Robotique.

Formé à Polytech'Montpellier, Je suis un professionnel de conception de projets logiciel applicatif ou embarqué dans les secteurs de l'Aéronautique, de la Robotique, des Drones et de la Vision par Ordinateur. Aussi, Je reste ouvert à d'autres types de secteurs tels que l'Énergie et les Finances.

Les Sciences de l'Ingénieur sont au cœur du métier d'ingénieur. Sur le site **ReadyDev**, la Plateforme de Développement Continu, dont j'en suis le concepteur, vous trouverez des cours et des tutoriels adaptés aux sciences de l'ingénieur.

J'aime, Je partage.

Gérard KESSE

[GitHub](#) | [LinkedIn](#) | [SiteWeb](#)



Sommaire

Auteur.....	2
Sommaire	3
Introduction	5
Installation sous Windows avec WampServer	5
Téléchargements	5
Installation de Notepad++	5
Installation de WampServer	5
Google Search Console.....	7
Faire indexer un site internet	7
Supprimer un site internet	7
Ajouter un site internet.....	7
Configurer le fichier robots.txt d'un site internet	7
Configurer le sitemap d'un site internet	8
Afficher un message.....	9
Afficher un message	9
Communiquer avec le clavier	10
Lire des données à partir du clavier	10
Gérer le problème d'espace lors de la saisie de données	12
Gérer les conditions	14
Condition If (Si)	14
Condition Else (Sinon)	16
Condition Else If (Sinon Si)	18
Gérer les boucles	20
Boucle For (Pour Tout)	20
Boucle While (Tant Que)	22
Boucle Do While (Faire Tant Que)	24

Introduction

Le PHP est un langage de programmation orienté objet utilisé côté serveur pour la création de pages web dynamiques. Le but de ce tutoriel est de vous apprendre le PHP.

Prérequis :

Aucun prérequis n'est nécessaire.

Installation sous Windows avec WampServer

Téléchargements

Notepad++ :

<https://notepad-plus-plus.org/fr/>

WampServer :

<http://www.wampserver.com/>

Installation de Notepad++

Plugins Notepad++ :

TextFX

NppExport

Installation de WampServer

Configurer un port Apache :

C:\wamp\bin\apache\apache2.4.9\conf\httpd.conf

Listen 0.0.0.0:80

Listen [::]:80

Activer l'utilisation de machines virtuelles Apache :

C:\wamp\bin\apache\apache2.4.9\conf\httpd.conf

Virtual hosts

Include conf/extra/httpd-vhosts.conf

Configurer une machine virtuelle Apache :

C:\wamp\bin\apache\apache2.4.9\conf\extra\httpd-vhosts.conf

```
<VirtualHost *:80>
```

```
    ServerName www.readydev.com
```

```
    ServerAdmin gerard.kesse@outlook.fr
```

```
    DocumentRoot "C:/Users/gerar/Programs/ReadyStudio/ReadyDev"
```

```
    <Directory "C:/Users/gerar/Programs/ReadyStudio/ReadyDev">
```

```
        Options All
```

```
        AllowOverride All
```

```
        Order allow,deny
```

```
        allow from all
```

```
        Require all granted
```

```
    </Directory>
```

```
</VirtualHost>
```

Activer l'utilisation de la réécriture d'URL :

WampServer

Clic gauche

Apache

Apache modules

rewrite_module

Mettre le serveur Apache en ligne :

WampServer

Clic gauche

Put Online

Tester l'utilisation du port 80 Apache :

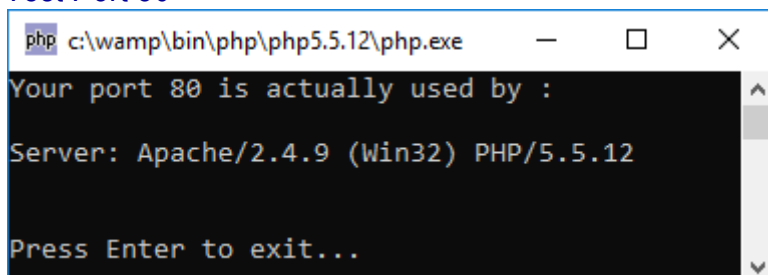
WampServer

Clic gauche

Apache

Service

Test Port 80



Google Search Console

Google Search Console est un service gratuit de Google qui permet aux webmasters de vérifier si leur site est indexé par Google et d'optimiser son référencement.

Faire indexer un site internet

Faire indexer un site internet :

<https://www.google.com/webmasters/tools/submit-url>

URL

<http://readydev.freeboxos.fr/>

Envoyer la demande

Supprimer un site internet

Supprimer un site internet :

<https://www.google.com/webmasters/tools/home?hl=fr>

Gérer une propriété

Supprimer la propriété

Ajouter un site internet

Ajouter un site internet :

<https://www.google.com/webmasters/tools/home?hl=fr>

Site web

<http://readydev.freeboxos.fr/>

Ajouter une propriété

Configurer le fichier robots.txt d'un site internet

Installer le fichier robots.txt d'un site internet :

<http://readydev.freeboxos.fr/robots.txt>

```
# Sitemaps
Sitemap: http://readydev.freeboxos.fr/data/sitemaps/sitemaps.xml

# Allow All
User-agent: *
Allow: /
```

Configurer le fichier robots.txt d'un site internet :<https://www.google.com/webmasters/tools/home?hl=fr><http://readydev.freeboxos.fr/>

Exploration

Outil de test du fichier robots.txt

Envoyer

Configurer le sitemap d'un site internet**Installer le sitemap d'un site internet :**<http://readydev.freeboxos.fr/data/sitemaps/sitemaps.xml>

```
- <sitemapindex xsi:schemaLocation="http://www.sitemaps.org/schemas/sitemap/0.9 http://www.sitemaps.org/schemas/sitemap/0.9/sitemap.xsd">
  - <sitemap>
    - <loc>
      http://readydev.freeboxos.fr/data/sitemaps/sitemap.xml
    </loc>
    <lastmod>2018-04-11</lastmod>
  </sitemap>
</sitemapindex>
```

Tester le sitemap d'un site internet :<https://www.google.com/webmasters/tools/home?hl=fr><http://readydev.freeboxos.fr/>

Exploration

Sitemaps

Ajouter/Tester un sitemap

data/sitemaps/sitemaps.xml

Test

Afficher le résultat de test

Fermer le test

Configurer le sitemap d'un site internet :<https://www.google.com/webmasters/tools/home?hl=fr><http://readydev.freeboxos.fr/>

Exploration

Sitemaps

Ajouter/Tester un sitemap

data/sitemaps/sitemaps.xml

Envoyer

Actualiser la page

Afficher un message

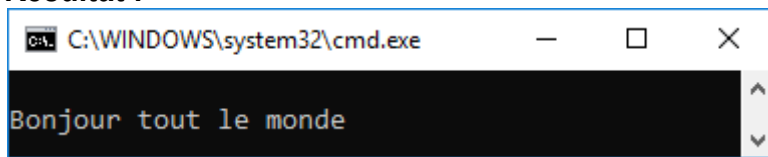
Afficher un message

Objectif :

Afficher un message dans la console.

Implémentation :

Afficher un message dans la console (Bonjour tout le monde).

Résultat :**Dossier projet :**

src/main.cpp
win/bin/
win/build/

src/main.cpp

```
//=====
#include <iostream>
//=====
using namespace std;
//=====
int main(int argc, char** argv) {
    cout << "Bonjour tout le monde\n";
    return 0;
}
//=====
```

Communiquer avec le clavier

Lire des données à partir du clavier

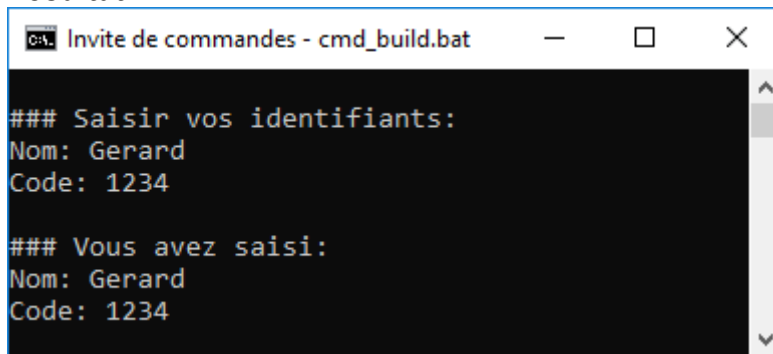
Objectif :

Récupérer des données saisies à partir du clavier.

Implémentation :

Récupérer les identifiants d'une personne saisis à partir du clavier.

Récupérer le nom (m_name) et le code d'entrée (m_code) de la personne.

Résultat :

```
Invite de commandes - cmd_build.bat

### Saisir vos identifiants:
Nom: Gerard
Code: 1234

### Vous avez saisi:
Nom: Gerard
Code: 1234
```

Dossier projet :

src/main.cpp

win/bin/

win/build/

src/main.cpp

```
//=====
#include <iostream>
//=====
using namespace std;
//=====
int main(int argc, char** argv) {
    string m_name;
    int m_code;

    cout << "\n### Saisir vos identifiants:\n";
    cout << "Nom: "; cin >> m_name;
    cout << "Code: "; cin >> m_code;

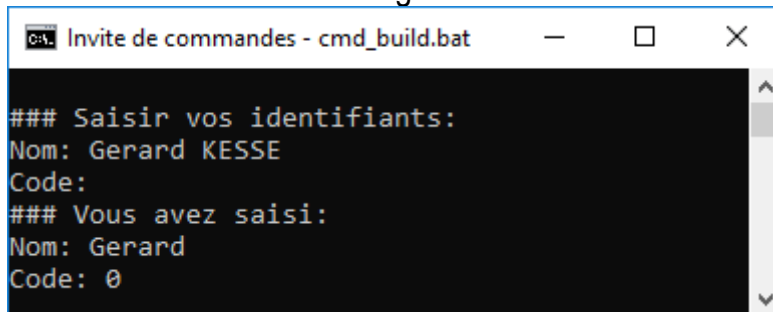
    cout << "\n### Vous avez saisi:\n";
    cout << "Nom: " << m_name << "\n";
    cout << "Code: " << m_code << "\n";
    return 0;
}
//=====
```

Problème :

Problème de gestion d'espace lors de la saisie

Nom : Gerard KESSE

Code : la saisie du code est ignorée



Gérer le problème d'espace lors de la saisie de données

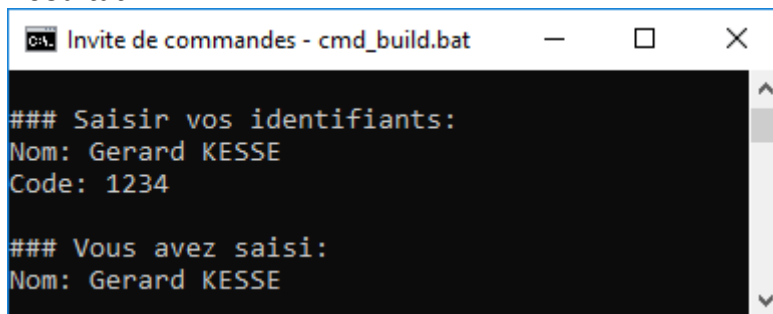
Objectif :

Gérer le problème d'espace lors de la saisie de données à partir du clavier.

Implémentation :

Récupérer les identifiants d'une personne saisis à partir du clavier.

Récupérer le nom (m_name) et le code d'entrée (m_code) de la personne.

Résultat :

```
### Saisir vos identifiants:
Nom: Gerard KESSE
Code: 1234

### Vous avez saisi:
Nom: Gerard KESSE
```

Dossier projet :

src/main.cpp

win/bin/

win/build/

src/main.cpp

```
//=====
#include <iostream>
#include <string>
//=====
using namespace std;
//=====
int main(int argc, char** argv) {
    string m_cin;
    string m_name;
    int m_code;

    cout << "\n### Saisir vos identifiants:\n";
    cout << "Nom: "; getline(cin, m_cin); m_name =
m_cin;
    cout << "Code: "; getline(cin, m_cin); m_code =
stoi(m_cin);

    cout << "\n### Vous avez saisi:\n";
    cout << "Nom: " << m_name << "\n";
    cout << "Code: " << m_code << "\n";
    return 0;
}
//=====
```

Gérer les conditions

Condition If (Si)

Objectif :

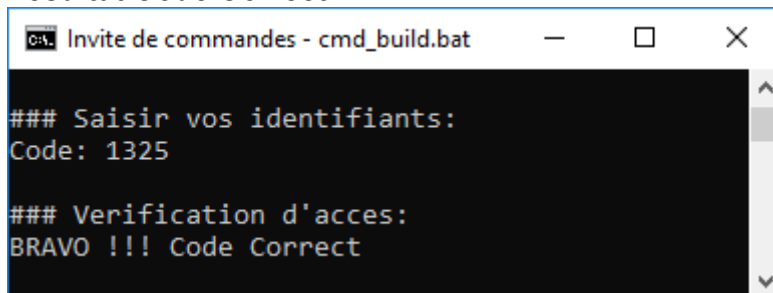
Exécuter une instruction Si une condition est vérifiée.

Implémentation :

Récupérer le code d'accès (m_code) d'une personne.

Afficher un message (BRAVO !!! Code Correct) si le code d'accès est correct.

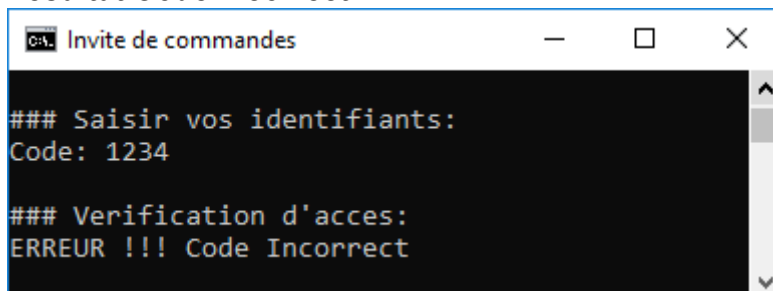
Résultat Code Correct :



```
### Saisir vos identifiants:
Code: 1325

### Verification d'accès:
BRAVO !!! Code Correct
```

Résultat Code Incorrect :



```
### Saisir vos identifiants:
Code: 1234

### Verification d'accès:
ERREUR !!! Code Incorrect
```

Dossier projet :

src/main.cpp

win/bin/

win/build/

src/main.cpp

```
//=====
#include <iostream>
#include <string>
//=====
using namespace std;
//=====
int main(int argc, char** argv) {
    string m_cin;
    int m_code;
    int ACCESS_CODE = 1325;

    cout << "\n### Saisir vos identifiants:\n";
    cout << "Code: "; getline(cin, m_cin); m_code =
stoi(m_cin);

    cout << "\n### Verification d'accès:\n";
    if(m_code == ACCESS_CODE) {
        cout << "BRAVO !!! Code Correct\n";
        return 0;
    }
    cout << "ERROR !!! Code Incorrect\n";
    return 0;
}
//=====
```

Condition Else (Sinon)

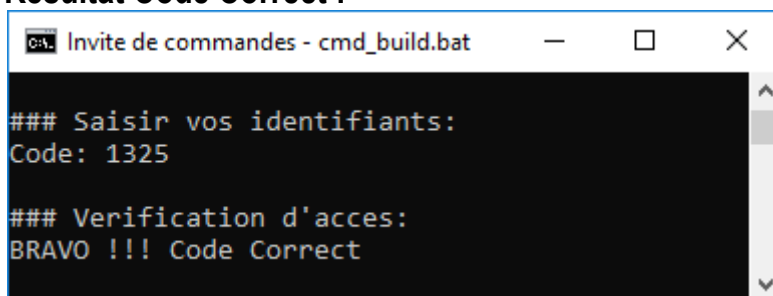
Objectif :

Exécuter une instruction Si une condition est vérifiée,
Sinon exécuter l'autre instruction.

Implémentation :

Récupérer le code d'accès (m_code) d'une personne.
Afficher un message (BRAVO !!! Code Correct) si le code d'accès est correct.
Sinon afficher le message (ERREUR !!! Code Incorrect).

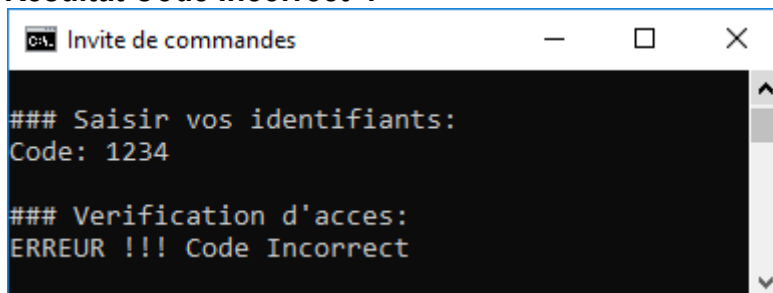
Résultat Code Correct :



```
### Saisir vos identifiants:
Code: 1325

### Verification d'accès:
BRAVO !!! Code Correct
```

Résultat Code Incorrect :



```
### Saisir vos identifiants:
Code: 1234

### Verification d'accès:
ERREUR !!! Code Incorrect
```

Dossier projet :

src/main.cpp
win/bin/
win/build/

src/main.cpp

```
//=====
#include <iostream>
#include <string>
//=====
using namespace std;
//=====
int main(int argc, char** argv) {
    string m_cin;
    int m_code;
    int ACCESS_CODE = 1325;

    cout << "\n### Saisir vos identifiants:\n";
    cout << "Code: "; getline(cin, m_cin); m_code =
stoi(m_cin);

    cout << "\n### Verification d'accès:\n";
    if(m_code == ACCESS_CODE) {
        cout << "BRAVO !!! Code Correct\n";
        return 0;
    }
    cout << "ERROR !!! Code Incorrect\n";
    return 0;
}
//=====
```

Condition Else If (Sinon Si)

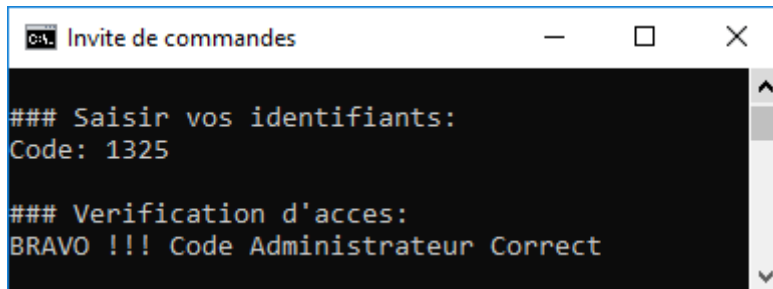
Objectif :

Exécuter une instruction Si une condition est vérifiée,
Sinon Si la condition suivante est vérifiée, exécuter l'autre instruction.

Implémentation :

Récupérer le code d'accès (m_code) d'une personne.
Si le code d'accès correspond au code d'accès administrateur,
Afficher un message (BRAVO !!! Code Administrateur Correct).
Sinon Si le code d'accès correspond au code d'accès utilisateur,
Afficher le message (BRAVO !!! Code Utilisateur Correct).
Dans le Cas Contraire (Sinon),
Afficher le message (ERREUR !!! Code Incorrect).

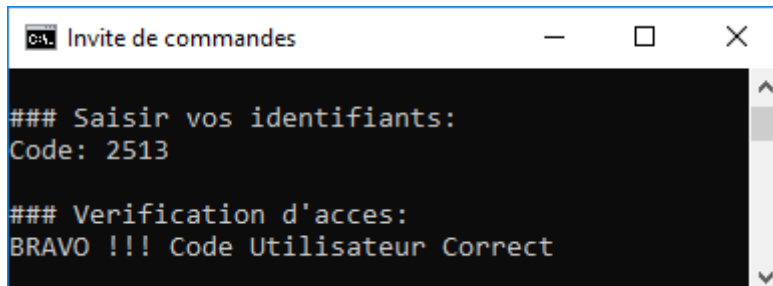
Résultat Code Administrateur Correct :



```
### Saisir vos identifiants:
Code: 1325

### Verification d'accès:
BRAVO !!! Code Administrateur Correct
```

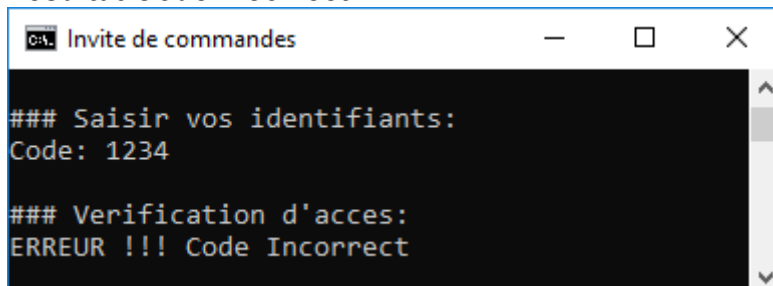
Résultat Code Utilisateur Correct :



```
### Saisir vos identifiants:
Code: 2513

### Verification d'accès:
BRAVO !!! Code Utilisateur Correct
```

Résultat Code Incorrect :



```
### Saisir vos identifiants:
Code: 1234

### Verification d'accès:
ERREUR !!! Code Incorrect
```

Dossier projet :

src/main.cpp
win/bin/
win/build/

src/main.cpp

```
//=====
#include <iostream>
#include <string>
//=====
using namespace std;
//=====
int main(int argc, char** argv) {
    string m_cin;
    int m_code;
    int ADMIN_CODE = 1325;
    int USER_CODE = 2513;

    cout << "\n### Saisir vos identifiants:\n";
    cout << "Code: "; getline(cin, m_cin); m_code =
stoi(m_cin);

    cout << "\n### Verification d'accès:\n";
    if(m_code == ADMIN_CODE) {
        cout << "BRAVO !!! Code Administrateur
Correct\n";
    }
    else if(m_code == USER_CODE) {
        cout << "BRAVO !!! Code Utilisateur
Correct\n";
    }
    else {
        cout << "ERREUR !!! Code Incorrect\n";
    }
    return 0;
}
//=====
```

Gérer les boucles

Boucle For (Pour Tout)

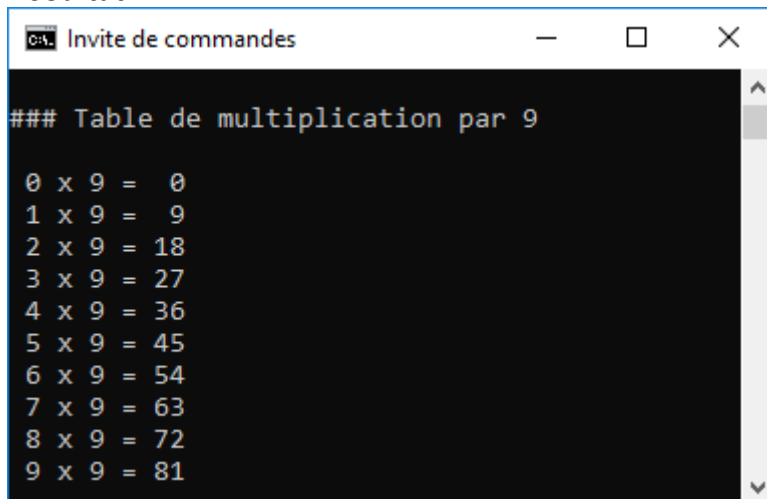
Objectif :

Exécuter une instruction en boucle.

Implémentation :

Créer une table de multiplication par 9.

Les lignes allant de (0 à 10).

Résultat :

```
### Table de multiplication par 9
0 x 9 = 0
1 x 9 = 9
2 x 9 = 18
3 x 9 = 27
4 x 9 = 36
5 x 9 = 45
6 x 9 = 54
7 x 9 = 63
8 x 9 = 72
9 x 9 = 81
```

Dossier projet :

src/main.cpp

win/bin/

win/build/

src/main.cpp

```
//=====
#include <iostream>
#include <iomanip>
//=====
using namespace std;
//=====
int main(int argc, char** argv) {
    int ROW_MAX = 10;
    int FACTOR = 9;

    cout << "### Table de multiplication par 9\n\n";

    for(int i = 0; i <= ROW_MAX; i++) {
        int m_data = i * FACTOR;
        cout << setw(2) << i << " x ";
        cout << FACTOR << " = ";
        cout << setw(2) << m_data << "\n";
    }
    return 0;
}
//=====
```

Boucle While (Tant Que)

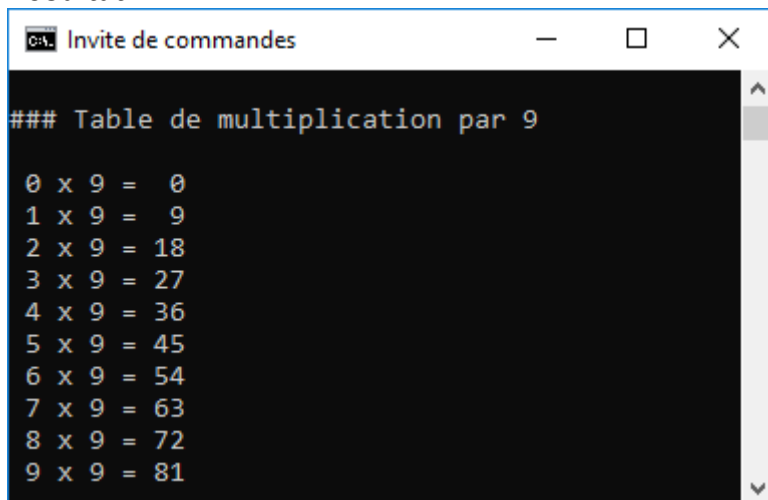
Objectif :

Exécuter une instruction en boucle.

Implémentation :

Créer une table de multiplication par 9.

Les lignes allant de (0 à 10).

Résultat :

```
### Table de multiplication par 9

0 x 9 = 0
1 x 9 = 9
2 x 9 = 18
3 x 9 = 27
4 x 9 = 36
5 x 9 = 45
6 x 9 = 54
7 x 9 = 63
8 x 9 = 72
9 x 9 = 81
```

Dossier projet :

src/main.cpp

win/bin/

win/build/

src/main.cpp

```
//=====
#include <iostream>
#include <iomanip>
//=====
using namespace std;
//=====
int main(int argc, char** argv) {
    int ROW_MAX = 10;
    int FACTOR = 9;

    cout << "### Table de multiplication par 9\n\n";

    int i = 0;

    while(i <= ROW_MAX) {
        int m_data = i * FACTOR;
        cout << setw(2) << i << " x ";
        cout << FACTOR << " = ";
        cout << setw(2) << m_data << "\n";
        i++;
    }
    return 0;
}
//=====
```

Boucle Do While (Faire Tant Que)

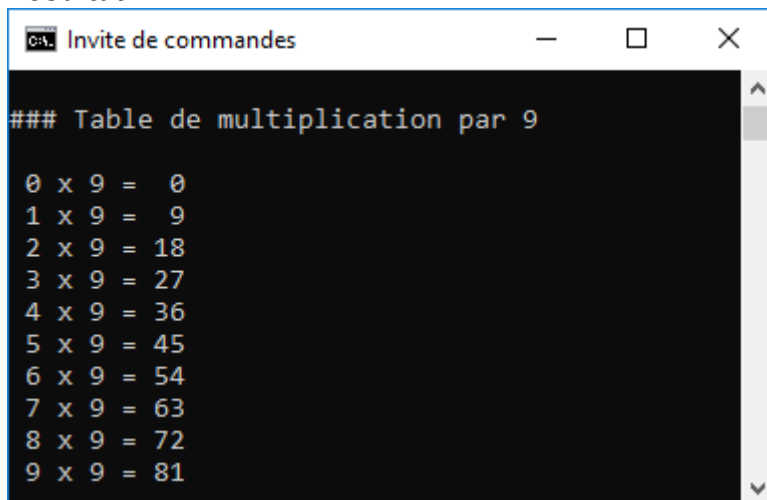
Objectif :

Exécuter une instruction en boucle.

Implémentation :

Créer une table de multiplication par 9.

Les lignes allant de (0 à 10).

Résultat :

```
### Table de multiplication par 9
0 x 9 = 0
1 x 9 = 9
2 x 9 = 18
3 x 9 = 27
4 x 9 = 36
5 x 9 = 45
6 x 9 = 54
7 x 9 = 63
8 x 9 = 72
9 x 9 = 81
```

Dossier projet :

src/main.cpp

win/bin/

win/build/

src/main.cpp

```
//=====
#include <iostream>
#include <iomanip>
//=====
using namespace std;
//=====
int main(int argc, char** argv) {
    int ROW_MAX = 10;
    int FACTOR = 9;

    cout << "### Table de multiplication par 9\n\n";

    int i = 0;

    do {
        int m_data = i * FACTOR;
        cout << setw(2) << i << " x ";
        cout << FACTOR << " = ";
        cout << setw(2) << m_data << "\n";
        i++;
    } while(i <= ROW_MAX);
    return 0;
}
//=====
```