

lesson 32 CPP Generic Programming

فى بداية كورس C++، تحدثنا أن لغة C++ قدمت لنا ميزتين وهما :
object oriented programming, generic programming

سنتحدث فى هذا الدرس عن **generic programming**
هل نتذكر فى كورس C، عند كتابة أمر printf، كنا نستخدم علامة % لتحديد نوع المتغير، %d إذا كان رقم، %f إذا كان رقم عشري، %c إذا كان حرف، %s إذا كان كلمة

كان لابد أن نعلم من المستخدم ماذا سيقوم بطباعته
لكن فى لغة C++، عند الطباعة نستخدم cout ثم اسم المتغير بدون تحديده للكمبيوتر، هو تلقائياً يقوم بتحديد، إذا ما كان char, int, string, float
وهذا ما يقوم به **generic programming**، يقوم calss أو function بمعرفة نوع المتغير وتتعامل معه بصورة طبيعية، لكن كيف نقوم بعمل function، مثل cout

نريد عمل function لتحديد أكبر رقم وسيكون كالتالى :

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

```
void printMax(int x, int y) {
    //function قمنا بعمل
    int z = x > y ? x : y;
    // قمنا بتحديد أى منهم الأكبر
    cout << z << endl;
    //نقوم بطباعة z
    return;
```

```
}
```

```
int main(){  
    printMax(6, 5);  
    //function نقوم بتنفيذ  
}
```

output:

6

إما إذا وضعنا float فسوف يعتبرها int أيضاً كالتالي:

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

```
void printMax(int x, int y) {  
    //function قمنا بعمل  
    int z = x > y ? x : y;  
    // قمنا بتحديد أى منهم الأكبر //  
    cout << z << endl;  
    //z نقوم بطباعة  
    return;  
}
```

```
int main(){  
    printMax(6.5, 5.9);  
    //function نقوم بتنفيذ  
}
```

output:

6

إما إذا وضعنا char فسوف يقوم بطباعة رقم أيضاً:

```

#include<iostream>
using namespace std;

void printMax(int x, int y) {
    //function قمنا بعمل
    int z = x > y ? x : y;
    // قمنا بتحديد أى منهم الأكبر
    cout << z << endl;
    //نقوم بطباعة z
    return;
}

```

```

int main(){
    printMax('G', 'T');
    //function نقوم بتنفيذ
}

```

output:

84

لكن باستخدام **generic programming**
 سيتعامل مع كل data types :

```

#include<iostream>
using namespace std;

template <typename gammal>
// بهذه الطريقة نقوم باستخدام generic programming
//data type gammal هنا يمثل template لأى
void printMax(gammal x, gammal y) {
    gammal z = x > y ? x : y;
}

```

```
// قمنا بتحديد أى منهم الأكبر
cout << z << endl;
// نقوم بطباعة z
return;
}

int main() {
    printMax(6, 5);
    printMax('G', 'T');
    printMax("Gammal", "Tech");
    //function نقوم بتنفيذ
}
```

output:

6

T

Tech