Statements in C++: A Summary

2-28-2025

by Atomic Badger

“What is a statement in the programming language C++? What are the most common types of statements in C++ programming? Give one simple example of each type of statement.”

ChatGPT response:

What is a Statement in C++?

A **statement** in C++ is a single instruction that tells the computer to perform an action. It is typically terminated with a **semicolon (;)** and forms the building blocks of a C++ program.

Most Common Types of Statements in C++

1. **Expression Statements**
2. **Declaration Statements**
3. **Control Statements**
4. **Loop Statements**
5. **Jump Statements**

---

1. Expression Statement

An **expression statement** performs an operation, such as assigning a value or calling a function.

Example:

```
int x = 10;  // Assignment statement
x += 5;      // Updating x
std::cout << x;  // Function call statement
```

---

2. Declaration Statement

A **declaration statement** is used to declare variables, constants, or functions.

Example:

```
int a = 5;  // Variable declaration
```

```
const double PI = 3.14159;  // Constant declaration
```

---

3. Control Statement

A **control statement** directs the flow of execution using **if, if-else, switch** statements.

Example (if-else statement):

```
int num = 10;
if (num > 0) {
    std::cout << "Positive number";
} else {
    std::cout << "Non-positive number";
}
```

Example (switch statement):

```
char grade = 'A';
switch (grade) {
    case 'A': std::cout << "Excellent"; break;
    case 'B': std::cout << "Good"; break;
    default: std::cout << "Try again";
}
```

---

4. Loop Statement

A **loop statement** allows repeated execution of a block of code (**for, while, do-while** loops).

Example (for loop):

```
for (int i = 1; i <= 5; i++) {
    std::cout << i << " ";
}
```

Example (while loop):

```
int i = 1;
while (i <= 5) {
    std::cout << i << " ";
    i++;
}
```

---

5. Jump Statement

A **jump statement** alters the normal flow of execution (**break, continue, return**).

Example (break statement in a loop):

```
for (int i = 1; i <= 10; i++) {
    if (i == 5) break;  // Exit loop when i is 5
    std::cout << i << " ";
}
```

Example (continue statement in a loop):

```
for (int i = 1; i <= 5; i++) {
    if (i == 3) continue;  // Skip iteration when i is 3
    std::cout << i << " ";
}
```

Example (return statement in a function):

```
int add(int a, int b) {
    return a + b;  // Returns the sum of a and b
}
```