C++ Constructors

By: Dr. Shweta Jain

Head Coordinator, Data Science

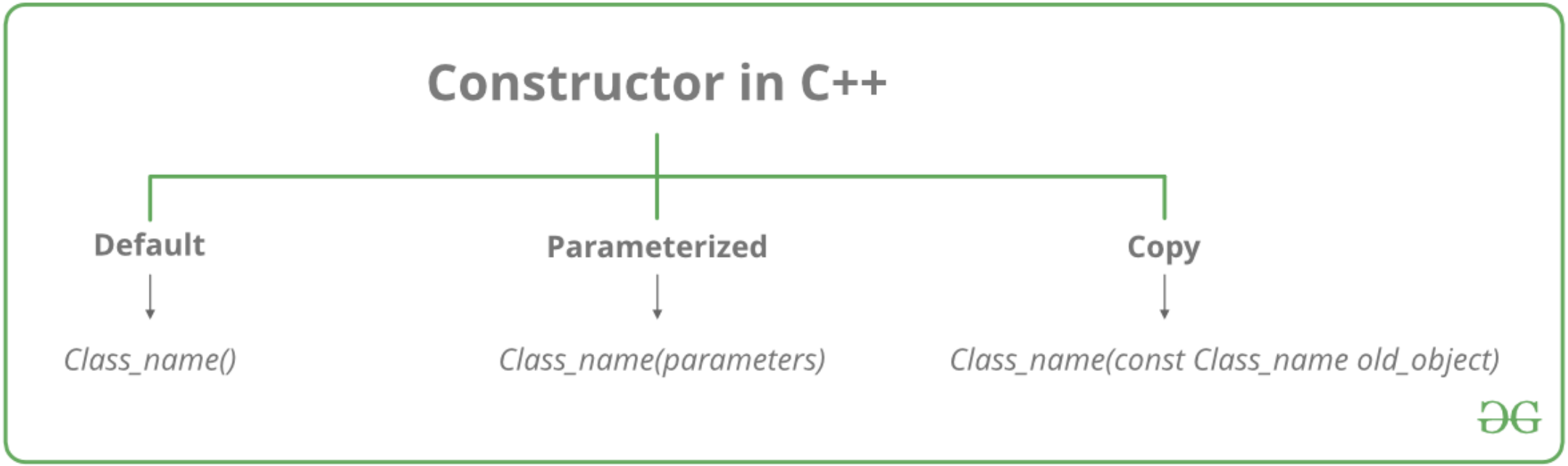
C++ Constructors

- A constructor is a special type of member function that is called automatically when an object is created.

- In C++, a constructor has the same name as that of the class and it does not have a return type. For example:

```
class  Wall {
  public:
    // create a constructor
    Wall() {
      // code
    }
};
```

C++ Constructors



C++ Default Constructor

- A constructor with no parameters is known as a default constructor. In the example above, Wall() is a default constructor.

```
// C++ program to demonstrate the use of default constructor

#include <iostream>
using namespace std;

// declare a class
class  Wall {
  private:
    double length;
public:
    // default constructor to initialize variable
    Wall() {
      length = 5.5;
      cout << "Creating a wall." << endl;
      cout << "Length = " << length << endl;
    }
};

int main() {
  Wall wall1;
  return 0;
}
```

Output:

Creating a Wall
Length = 5.5

C++ Parameterized Constructor

- In C++, a constructor with parameters is known as a parameterized constructor. This is the preferred method to initialize member data.

```
// C++ program to calculate the area of a wall

#include <iostream>
using namespace std;

// declare a class
class Wall {
  private:
    double length;
    double height;

  public:
    // parameterized constructor to initialize variables
    Wall(double len, double hgt) {
      length = len;
      height = hgt;
    }
    double calculateArea() {
      return length * height;
    }
};

int main() {
  // create object and initialize data members
  Wall wall1(10.5, 8.6);
  Wall wall2(8.5, 6.3);

  cout << "Area of Wall 1: " << wall1.calculateArea() << endl;
  cout << "Area of Wall 2: " << wall2.calculateArea();

  return 0;
}
```

Output

```
Area of Wall 1: 90.3
Area of Wall 2: 53.55
```

C++ Copy Constructor

- The copy constructor in C++ is used to copy data of one object to another.

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

// declare a class
class Wall {
  private:
    double length;
    double height;

  public:
    // initialize variables with parameterized constructor
    Wall(double len, double hgt) {
      length = len;
      height = hgt;
    }

    // copy constructor with a Wall object as parameter
    // copies data of the obj parameter
    Wall(Wall &obj) {
      length = obj.length;
      height = obj.height;
    }
    double calculateArea() {
        return length * height;
    }
};

int main() {
  // create an object of Wall class
  Wall wall1(10.5, 8.6);

  // copy contents of wall1 to wall2
  Wall wall2 = wall1;

  // print areas of wall1 and wall2
  cout << "Area of Wall 1: " << wall1.calculateArea() << endl;
  cout << "Area of Wall 2: " << wall2.calculateArea();

  return 0;
}
```

Output:

```
Area of Wall 1: 90.3
Area of Wall 2: 90.3
```