Tutorial 21 - Classes, Public and Private Access Modifiers in C++

Why Use Classes Instead of Structures?

1. **Data Hiding**:
   - In structures, all members are public, making data security a concern.
   - Classes allow data hiding through access modifiers.
2. **Functionality**:
   - Structures cannot contain functions.
   - Classes combine data and functions into a single unit.

Classes in C++

- **Definition**: User-defined data types that serve as templates for creating objects.
- **Components**:
  - Variables and functions (collectively called class members).

Access Modifiers in C++

1. **Public**:
   - Members can be accessed both inside and outside the class.
   - Accessed using the dot (`.`) operator.
2. **Private**:
   - Members are accessible only within the class.
   - Cannot be accessed directly by objects or outside functions.

Code Example

Class and Member Functions

```
class Employee {
private:
    int a, b, c; // Private members
public:
    int d, e;    // Public members
    void setData(int a1, int b1, int c1); // Declaration
    void getData() {
        cout << "The value of a is " << a << endl;
        cout << "The value of b is " << b << endl;
        cout << "The value of c is " << c << endl;
        cout << "The value of d is " << d << endl;
        cout << "The value of e is " << e << endl;
    }
};
// Definition of setData using scope resolution operator
void Employee::setData(int a1, int b1, int c1) {
    a = a1;
    b = b1;
    c = c1;
```

```
}

```

Main Function

```
int main() {
    Employee harry;
    // Assigning values to public members
    harry.d = 34;
    harry.e = 89;
    // Setting private member values using a public function
    harry.setData(1, 2, 4);
    // Printing all member values
    harry.getData();
    return 0;
}

```

Key Points from the Example

1. **Private Members**:
   - `a`, `b`, `c` are private and cannot be accessed directly.
   - Values are assigned through the public function `setData`.
2. **Public Members**:
   - `d`, `e` can be accessed and modified directly using the object.
3. **Scope Resolution Operator (`::`)**:
   - Used to define class functions outside the class.

Output of the Program

```
The value of a is 1
The value of b is 2
The value of c is 4
The value of d is 34
The value of e is 89

```

Short Notes

1. **Classes vs. Structures**:
   - **Structures**: All members are public; no data hiding or functions.
   - **Classes**: Allow data hiding and include functions.
2. **Access Modifiers**:
   - **Public**: Accessible everywhere.
   - **Private**: Accessible only within the class.
3. **Important Syntax**:
   - **Class Definition**:

     ```
     class ClassName {
     private:
         // Private members
     public:
     ```

```
    // Public members
};

```

- **Defining Functions Outside Class**:

```
void ClassName::FunctionName() {
    // Function definition
}

```

4. **Example Highlights**:
   - Use `setData` to assign private members.
   - Use `getData` to display all member values.