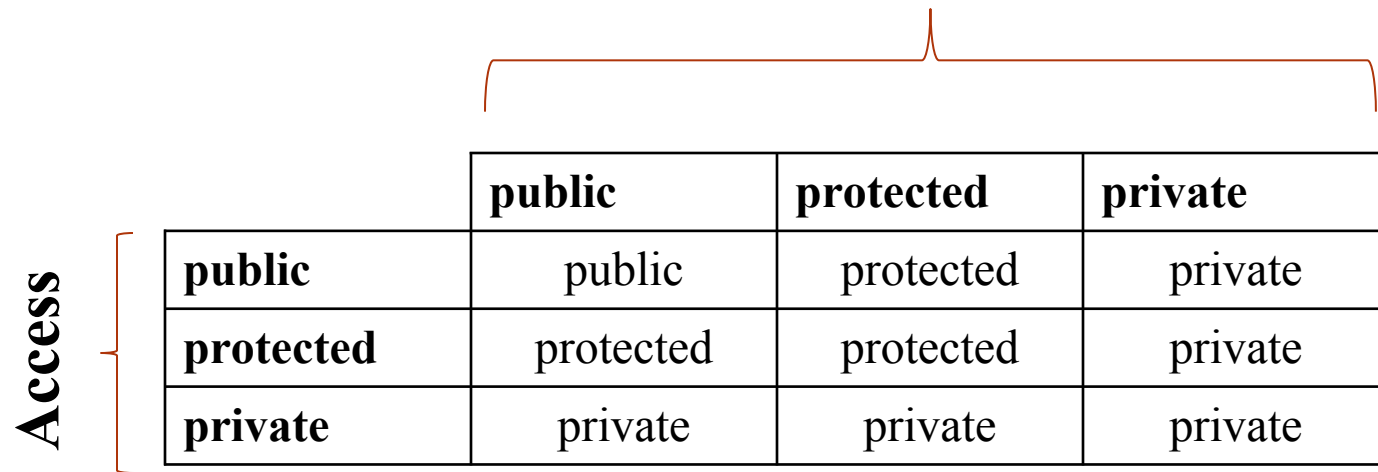


# Public, Protected, and Private Inheritance

# Inheritance access chart

## Inheritance



The diagram illustrates the inheritance access rules for a 3x3 matrix. A vertical bracket on the left, labeled 'Access', groups the rows. A horizontal bracket above the columns, labeled 'Inheritance', groups the columns. The matrix shows that public access is only inherited from public access, while protected and private access are inherited from all three access levels.

	<b>public</b>	<b>protected</b>	<b>private</b>
<b>public</b>	public	protected	private
<b>protected</b>	protected	protected	private
<b>private</b>	private	private	private

# Example 1

```
class A{
private: int priA;
protected: int proA;
public: int pubA;
    void displayA(){
        cout<<priA<<proA<<pubA;
    }
};

class B : public A{
private: int priB;
protected: int proB;
public: int pubB;
    void displayB(){
        cout<<priA<<proA<<pubA;
        cout<<priB<<proB<<pubB;
    }
};

int main(){
    A objA; B objB;
    cout<<objA.priA << objA.proA
    <<objA.pubA<<endl;
    cout<<objB.priA << objB.proA << objB.pubA
    <<endl;
    cout<<objB.priB << objB.proB
```

## Example 2

```
class A{
private: int priA;
protected: int proA;
public: int pubA;
    void display(){
        cout<<priA<<proA<<pubA;
    }
};

class B : protected A{
private: int priB;
protected: int proB;
public: int pubB;
    void display(){
        cout<<priA<<proA<<pubA;
        cout<<priB<<proB<<pubB;
    }
};

int main(){
    A objA; B objB;
    cout<<objA.priA << objA.proA
    <<objA.pubA<<endl;
    cout<<objB.priA <<objB.proA << objB.pubA
    <<endl;
    cout<<objB.priB << objB.proB
```

```
class A {
private: int priA;
protected: int proA;
public: int pubA;
    void display() {
        cout<<priA<<proA<<pubA;
    }
};

class B : private A {
private: int priB;
protected: int proB;
public: int pubB;
    void display() {
        cout<<priA<<proA<<pubA;
        cout<<priB<<proB<<pubB;
    }
};

int main() {
    A objA; B objB;
    cout<<objA.priA << objA.proA
    <<objA.pubA<<endl;
    cout<<objB.priA <<objB.proA << objB.pubA
    <<endl;
    cout<<objB.priB << objB.proB
```

```

class A{
private: int priA;
protected: int proA;
public: int pubA;
    void display(){
        cout<<priA<<proA<<pubA;
    }
};

```

```

class B : public A{
private: int priB;
protected: int proB;
public: int pubB;
    void display(){
        cout<<priA<<proA<<pubA;
        cout<<priB<<proB<<pubB;
    }
};

```

```

class C : public B{
private: int priC;
protected: int proC;
public: int pubC;
    void display(){
        cout<<priA<<proA<<pubA;
        cout<<priB<<proB<<pubB;
        cout<<priC<<proC<<pubC;
    }
};

```

```

int main(){
    A objA; B objB; C objC;
    cout<<objA.priA << objA.proA
    <<objA.pubA<<endl;
    cout<<objB.priA << objB.proA
    <<objB.pubA<<endl;
    cout<<objB.priB << objB.proB
    <<objB.pubB<<endl;
    cout<<objC.priA << objC.proA
    <<objC.pubA<<endl;
    cout<<objC.priB << objC.proB
    <<objC.pubB<<endl;
    cout<<objC.priC << objC.proC
    <<objC.pubC<<endl;
}

```

**Example**

**4**

```

class A {
private: int priA;
protected: int proA;
public: int pubA;
    void display() {
        cout << priA << proA << pubA;
    }
};

```

```

class B : protected A {
private: int priB;
protected: int proB;
public: int pubB;
    void display() {
        cout << priA << proA << pubA;
        cout << priB << proB << pubB;
    }
};

```

```

class C : public B {
private: int priC;
protected: int proC;
public: int pubC;
    void display() {
        cout << priA << proA << pubA;
        cout << priB << proB << pubB;
        cout << priC << proC << pubC;
    }
};

```

```

int main() {
    A objA; B objB; C objC;
    cout << objA.priA << objA.proA
    << objA.pubA << endl;
    cout << objB.priA << objB.proA
    << objB.pubA << endl;
    cout << objB.priB << objB.proB
    << objB.pubB << endl;
    cout << objC.priA << objC.proA
    << objC.pubA << endl;
    cout << objC.priB << objC.proB
    << objC.pubB << endl;
    cout << objC.priC << objC.proC
    << objC.pubC << endl;
}

```

**Example**

**5**

```

class A{
private: int priA;
protected: int proA;
public: int pubA;
    void display(){
        cout<<priA<<proA<<pubA;
    }
};

```

```

class B : private A{
private: int priB;
protected: int proB;
public: int pubB;
    void display(){
        cout<<priA<<proA<<pubA;
        cout<<priB<<proB<<pubB;
    }
};

```

```

class C : public B{
private: int priC;
protected: int proC;
public: int pubC;
    void display(){
        cout<<priA<<proA<<pubA;
        cout<<priB<<proB<<pubB;
        cout<<priC<<proC<<pubC;
    }
};

```

```

int main(){
    A objA; B objB; C objC;
    cout<<objA.priA << objA.proA
    <<objA.pubA<<endl;
    cout<<objB.priA << objB.proA
    <<objB.pubA<<endl;
    cout<<objB.priB << objB.proB
    <<objB.pubB<<endl;
    cout<<objC.priA << objC.proA
    <<objC.pubA<<endl;
    cout<<objC.priB << objC.proB
    <<objC.pubB<<endl;
    cout<<objC.priC << objC.proC
    <<objC.pubC<<endl;
}

```

**Example**

**6**