Refer Inheritance file (.cpp)

The two code snippets you provided demonstrate different forms of inheritance in C++: public inheritance and protected inheritance. The difference between them lies in how the accessibility of the base class members is altered in the derived class.

* **Inheritance Type:** Public
* **Access Control:**
* Protected members of Vehicle2 (e.g., brand and honk) remain protected in Car2.
* The derived class Car2 can access these protected members and expose them via its public methods.

 **Inheritance Type:** Protected

 **Access Control:**

* Public and protected members of Vehicle2 become protected members in Car2.
* The derived class Car2 can access these members and expose them via its public methods

**Key Differences:**

1. **Member Accessibility in Derived Class:**
   * **Public Inheritance:** The protected members of Vehicle2 (brand and honk) remain protected in Car2. If there were public members in Vehicle2, they would remain public in Car2.
   * **Protected Inheritance:** Both public and protected members of Vehicle2 become protected members in Car2.
2. **Visibility to Outside Code:**
   * **Public Inheritance:** If Vehicle2 had public members, they would be accessible from an object of Car2. Since brand and honk are protected, they are not directly accessible, but can be accessed through public methods of Car2.
   * **Protected Inheritance:** No members of Vehicle2 are accessible from an object of Car2. All inherited members are protected, and access must be provided through public methods of Car2.

In both examples, the method2 function works correctly because Car2 provides public methods (useHonk and getBrand) to access the protected members of Vehicle2. The difference would be more noticeable if Vehicle2 had public members or if external code tried to access the inherited members directly.