drill.cpp – drill упражнението от Chapter 14

Drill

Unfortunately, we can't construct a drill for the understanding of general design principles, so here we focus on the language features that support object-oriented programming.

1. Define a class **B1** with a virtual function **vf()** and a non-virtual function **f()**. Define both of these functions within class **B1**. Implement each function to output its name (e.g., **B1::vf()**). Make the functions **public**. Make a **B1** object and call each function.
2. Derive a class **D1** from **B1** and override **vf()**. Make a **D1** object and call **vf()** and **f()** for it.
3. Define a reference to **B1** (a **B1&**) and initialize that to the **D1** object you just defined. Call **vf()** and **f()** for that reference.
4. Now define a function called **f()** for **D1** and repeat 1–3. Explain the results.
5. Add a pure virtual function called **pvf()** to **B1** and try to repeat 1–4. Explain the result.
6. Define a class **D2** derived from **D1** and override **pvf()** in **D2**. Make an object of class **D2** and invoke **f()**, **vf()**, and **pvf()** for it.
7. Define a class **B2** with a pure virtual function **pvf()**. Define a class **D21** with a **string** data member and a member function that overrides **pvf()**; **D21::pvf()** should output the value of the **string**. Define a class **D22** that is just like **D21** except that its data member is an **int**. Define a function **f()** that takes a **B2&** argument and calls **pvf()** for its argument. Call **f()** with a **D21** and a **D22**.