1. Write a program to create a reference that is not initialized when it is created. Change a reference to refer to another object after it is initialized. Create a NULL reference.

2. Write a function that takes a pointer argument, modifies what the pointer points to, and then returns the destination of the pointer as a reference.

3. Create a class with some member functions, and make that the object that is pointed to by the argument. Make the pointer a **const** and make some of the member functions **const** and prove that you can only call the **const** member functions inside your function. Make the argument to your function a reference instead of a pointer.

4. Create a function that takes an argument of a reference to a pointer to a pointer and modifies that argument. In **main ( )**, call the function.

5. Create a function that takes a **char&** argument and modifies that argument. In **main ( )**, print out a **char** variable, call your function for that variable, and print it out again to prove to yourself that it has been changed. How does this affect program readability?

6. Write a class that has a **const** member function and a non-**const** member function. Write three functions that take an object of that class as an argument; the first takes it by value, the second by reference, and the third by **const** reference. Inside the functions, try to call both member functions of your class and explain the results.