

# Assessment Solutions

## Chapter 19 – Using the Singleton Pattern

1. **a – c:** Please see `Assessments/Chapter19/Chp19-Q1.cpp`
2. We cannot label the `static instance()` method as `virtual` in `Singleton` and override it in `President` simply because static methods can never be `virtual`. They are statically bound and also never receive a `this` pointer. Also, the signature may need to be different (and no one likes an un-intentional function hiding situation).
3. Other examples which may easily incorporate the Singleton pattern include creating a Singleton `CEO` of a company or a Singleton `TreasuryDepartment` for a country or a Singleton `Queen` of a nation. Each of these Singleton instances offers the opportunity to establish a registry to keep track of multiple Singleton objects. That is, many countries may have a single `Queen`. In this case, the registry would allow not just one Singleton per object type, but one Singleton per other qualifiers, such as *nation*. This is an example of the rare case where more than one Singleton object of a given type can occur (but always a controlled number of such objects).