

The Firebase Manager class implements the Singleton design pattern. A singleton design pattern ensures that only one instance of a particular class is ever created and that there is only one way to access that instance globally. The Firebase Manager class ensures this by utilizing a couple features. First, all variables in the class are private and have getters and setters as necessary to access them. Additionally, the constructor for the class is private so that it can only be instantiated through the public getInstance() function. This function checks whether an instance exists, and only creates a new instance if a current instance does not exist. If an instance already exists, the original instance is returned. Through this getInstance() function, the class ensures that it uses the singleton design pattern such that only one instance is ever created and it can only be accessed one way.