

Task 3: Proxy Create a proxy class for accessing a sensitive object that contains a secret key. The proxy should only allow access to the secret key if a correct password is provided.**package** com.Day28;

```
public class ProxyPatternDemo {  
    // Interface for accessing the secret key  
    interface SecretKeyAccess {  
        String getSecretKey(String password);  
    }  
    // Real object that contains the secret key  
    static class RealSecretKey implements SecretKeyAccess {  
        private final String secretKey;  
        public RealSecretKey() {  
            // Initialize the secret key (for demonstration purposes, it's a hardcoded string)  
            this.secretKey = "ThisIsASecretKey123!";  
        }  
        @Override  
        public String getSecretKey(String password) {  
            // Direct access to the secret key (bypassed in real use cases)  
            return this.secretKey;  
        }  
    }  
    // Proxy class that controls access to the real object  
    static class SecretKeyProxy implements SecretKeyAccess {  
        private final RealSecretKey realSecretKey;  
        private final String correctPassword;  
        public SecretKeyProxy(String correctPassword) {  
            this.realSecretKey = new RealSecretKey();  
            this.correctPassword = correctPassword;  
        }  
        @Override  
        public String getSecretKey(String password) {  
            // Check the provided password  
            if (this.correctPassword.equals(password)) {  
                // Password is correct, grant access to the secret key  
                return realSecretKey.getSecretKey(password);  
            } else {  
                // Password is incorrect, deny access  
                return "Access Denied: Incorrect Password";  
            }  
        }  
    }  
}  
// Main method to demonstrate the proxy pattern  
public static void main(String[] args) {  
    // Correct password for accessing the secret key  
    String correctPassword = "password123";  
    // Create a proxy object with the correct password  
    SecretKeyProxy proxy = new SecretKeyProxy(correctPassword);  
    // Attempt to access the secret key with the correct password  
    System.out.println(proxy.getSecretKey("password123")); // Output: ThisIsASecretKey123!  
    // Attempt to access the secret key with an incorrect password  
    System.out.println(proxy.getSecretKey("wrongpassword")); // Output: Access Denied: Incorrect  
    Password
```

}
}