시퀀스다이어그램

1. 회원가입

:Guest → :SignUpUI : 1. selectSignUp()

:SignUpUI → :SignUpChecker : 2. showInputForm()

:Guest → :SignUpUI : 3. enterInfo(ID, PW, Pnum)

:SignUpUI → :SignUpChecker : 4. submitSignUpInfo(ID, PW, Pnum)

:SignUpChecker → :Member : 5. createAccount(ID, PW, Pnum)

:Member → :SignUpChecker : 6. success

:SignUpChecker → :SignUpUI : 7. showSuccessMessage()

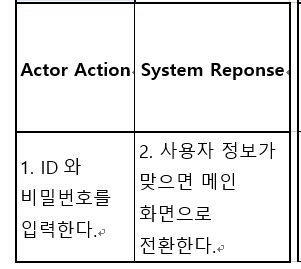
:SignUpUI → :Guest : 8. displaySuccess()

2. 로그인

actor : User

use case 명 : 로그인

requirement : 관리자와 사용자는 ID와 비밀번호로 로그인을 할 수 있어야 한다.



**등장 객체 (클래스 종류):**

* :User (<<actor>>)
* :LoginUI (<<boundary>>)
* :LoginChecker (<<control>>)
* :Account (<<entity>>)

:User → :LoginUI : 1. enterCredentials(ID, PW)

:LoginUI → :LoginChecker : 2. submitLoginInfo(ID, PW)

:LoginChecker → :Account : 3. validateCredentials(ID, PW)

:Account → :LoginChecker : 4. result(success/fail)

:LoginChecker → :LoginUI : 5. routeToMain() / showErrorMessage()

:LoginUI → :User : 6. displayResult()

3. 로그아웃

* :User (actor)
* :LogoutUI (<<boundary>>)
* :LogoutHandler (<<control>>)
* :SessionManager (<<entity>>)

 **Actor**: :User

 **Use Case 명**: 로그아웃

 **Requirement**: 관리자와 사용자는 시스템에서 로그아웃을 할 수 있어야 한다.

:User → :LogoutUI : 1. selectLogout()

:LogoutUI → :LogoutHandler : 2. requestLogout()

:LogoutHandler → :SessionManager : 3. terminateSession()

:SessionManager → :LogoutHandler : 4. sessionTerminated()

:LogoutHandler → :LogoutUI : 5. showLoginPage()

:LogoutUI → :User : 6. displayLoginPage()

4. 대여소 등록

actor : Admin

use case 명: 대여소 등록

Requirement : 관리자는 대여소의 이름, 위치, 보관 가능 수량, 운영 시간 정보를 입력하여 대여소를 등록할 수 있어야 한다.

descriptions : 