Final Project: Python Programming for Everybody (COSE156)

**1. 프로젝트 개요**

본 프로젝트는 파이썬을 이용하여 웹 브라우저로 접속 가능한 간단한 대화 서버를 작성하는 것으로써 사용자는 개별적인 계정을 생성하고 이를 통하여 로그인하여 다른 사용자와 대화를 나눌 수 있도록 하고 운영자는 기록의 덤프를 생성할 수 있는 시스템을 구축하는 것을 목표로 한다.

**2. 프로그램 실행 방법**

프로그램은 파이썬 프롬프트에서 app.py 파일을 실행하면 localhost의 8080번 포트에서 실행된다.

**3. 오류 상황 및 처리 방식**

가. 로그인 실패 시에는 로그인 실패 페이지로 이동한다.

나. 관리자 로그인 실패 시에는 관리자 로그인 실패 페이지로 이동한다.

- 로그인 실패 조건: ID와 PW의 조합이 등록되지 않은 경우

다. 계정 생성 실패 시에는 계정 생성 실패 페이지로 이동한다.

- 계정 생성 실패 조건: ID의 부재, PW의 부재, PW와 PW 확인이 불일치하는 경우

**4. 소스 코드 목록(전체)**

|  |  |
| --- | --- |
| Python | app.py: 파이썬 웹 서버 구동을 위한 소스 코드 |
| HTML | admin.html: 관리자 페이지  admin\_login.html: 관리자 페이지 로그인 페이지  admin\_logout.html: 관리자 페이지 로그아웃 페이지  confirm.html: 회원 등록 완료 페이지  error\_adminlogin.html: 관리자 페이지 로그인 실패 페이지  error\_login.html: 로그인 실패 페이지  error\_register: 회원 등록 실패 페이지  join.html: 대화 참가 페이지  logout.html: 로그아웃 완료 페이지  main.html: 로그인 페이지  register.html: 회원 등록 페이지  talk.html: 대화 페이지 |
| History File | talk.hist: 대화 로그 |
| Others | favicon.ico: 사이트 패비콘  html5.css: 사이트 스타일 시트 |

**5. 클래스별 설명(app.py)**

가. MainHandler: main.html 렌더링

나. JoinHandler: main.html에서 POST로 받은 로그인 정보를 바탕으로 로그인에 성공하면 join.html을 렌더링하고 그렇지 않으면 error\_login.html을 렌더링하여 로그인을 수행

다. RegisterHandler: register.html에서 POST로 받은 계정 정보를 바탕으로 계정 등록에 성공하면 confirm.html을 렌더링하고 그렇지 않으면 error\_register.html을 렌더링하여 계정 등록을 수행

라. TalkHandler: join.html에서 POST로 받은 세션 정보를 바탕으로 세션이 유효하면 talk.html을 렌더링하여 세션 정보를 검증하고 대화 로그를 남겨 대화 기능을 실질적으로 수행

마. AdminLoginHandler: admin\_login.html에서 POST로 받은 로그인 정보를 바탕으로 로그인에 성공하면 admin.html을 렌더링하고 그렇지 않으면 error\_adminlogin.html을 렌더링하여 관리자 페이지 로그인을 수행

바. AdminHandler:

바-A. AdminHandler.get(self): admin.html을 작동시키기 전에 관리자 계정으로 로그인된 세션이 있는지 점검하여 현재 사용자가 관리자가 맞는지 다시 한 번 확인함

바-B. AdminHandler.post(self): admin.html에서 POST로 받은 명령어를 바탕으로 talk.hist를 작성하며 이 과정에서도 위와 같이 관리자 계정으로 로그인된 세션이 있는지 점검함

사. LogoutHandler: talk.html에서 POST로 받은 세션 정보를 바탕으로 로그아웃 로그를 대화방에 남기고 해당 사용자의 세션을 삭제함

아. AdminLogoutHandler: admin.html에서 POST로 받은 관리자 세션 정보를 바탕으로 로그아웃 로그를 대화방에 남기지 않고 관리자의 세션을 삭제함

**6. HTML 파일명 및 사용된 메소드 이름**

가. HTML 파일명: 소스 코드 목록에 기재한 것과 같음

나. HTML에 관련하여 사용된 메소드 이름(app.py 클래스순):

나-A. main.html: MainHandler.get(self)

나-B. join.html: JoinHandler.post(self)

나-C. error\_login.html: JoinHandler.post(self)

나-D. confirm.html: RegisterHandler.post(self)

나-E. error\_register.html: RegisterHandler.post(self)

나-F. register.html: RegisterHandler.get(self)

나-G. talk.html: TalkHandler.post(self)

나-H. admin.html: AdminLoginHandler.post(self), AdminHandler.get(self),

AdminHandler.post(self)

나-I. error\_adminlogin.html: AdminLoginHandler.post(self)

나-J. admin\_login.html: AdminHandler.get(self), AdminHandler.post(self)

나-K. logout.html: LogoutHandler.post(self)

나-L. admin\_logout.html: AdminLogoutHandler.post(self)

다. 파이썬 웹 서버 구현에 관련하여 사용된 메소드 전문(app.py 클래스순)

|  |  |
| --- | --- |
| 다-A  다-B  다-C  다-D | class FastStop:  def \_\_init\_\_(self):  self.is\_closing = False  def enable(self):  def signal\_handler(signum, frame):  self.is\_closing = True  def try\_exit():  if self.is\_closing:  IOLoop.instance().stop()  signal.signal(signal.SIGINT, signal\_handler)  PeriodicCallback(try\_exit, 100).start() |
| 다-E | def make\_app():  handlers = []  handlers.append(url(r"/",MainHandler))  handlers.append(url(r'/join',JoinHandler))  handlers.append(url(r"/register",RegisterHandler))  handlers.append(url(r'/talk',TalkHandler))  handlers.append(url(r'/admin',AdminHandler))  handlers.append(url(r'/admin\_login',AdminLoginHandler))  handlers.append(url(r'/logout',LogoutHandler))  handlers.append(url(r"/files/(.\*)", StaticFileHandler, {"path": os.path.dirname(os.path.abspath(\_\_file\_\_))}))  return Application(handlers, static\_path = os.path.join(os.path.dirname(\_\_file\_\_), "files")) |

**7. 개발 목표에 대한 자기평가**

|  |  |  |
| --- | --- | --- |
| 가 | 로그인(대화 및 관리자 페이지) |  |
| 가-A | 로그인을 해야만 대화에 참여 가능한가? | Yes |
| 가-B | 등록된 계정에 대하여 로그인이 정상적으로 수행되는가? | Yes |
| 가-C | 등록되지 않은 계정에 대하여 로그인 실패 메시지가 나오는가? | Yes |
| 나 | 계정 등록 |  |
| 나-A | 유효한 ID와 PW를 바탕으로 계정 등록이 정상적으로 수행되는가? | Yes |
| 나-B | 유효하지 않은 ID와 PW에 대하여 계정 등록 실패 메시지가 나오는가? | Yes |
| 다 | 대화 |  |
| 다-A | 로그인에 성공한 사용자가 대화에 참여할 수 있는가? | Yes |
| 다-B | 대화에 참여할 때 대화 참여를 알리는 메시지가 발신되는가? | Yes |
| 다-C | 서로 다른 사용자들이 대화에 참여할 수 있으며 서로 식별되는가? | Yes |
| 라 | 로그아웃 |  |
| 라-A | 로그아웃을 하면 실제로 세션이 삭제되는가? | Yes |
| 라-B | 로그아웃을 하면 대화에 참여할 수 없게 되는가? | Yes |
| 마 | 관리자 페이지(기능) |  |
| 마-A | 관리자 페이지의 dump 및 load 기능이 작동하는가? | Yes |
| 바 | Challenge Problem |  |
| 바-A | 로그인을 해야만 관리자 페이지에 접속 가능한가? | Yes |
| 바-B | 관리자 페이지도 로그아웃이 가능한가? | Yes |
| 전체 | 개발 완성도 | 100% |