

1. 목적

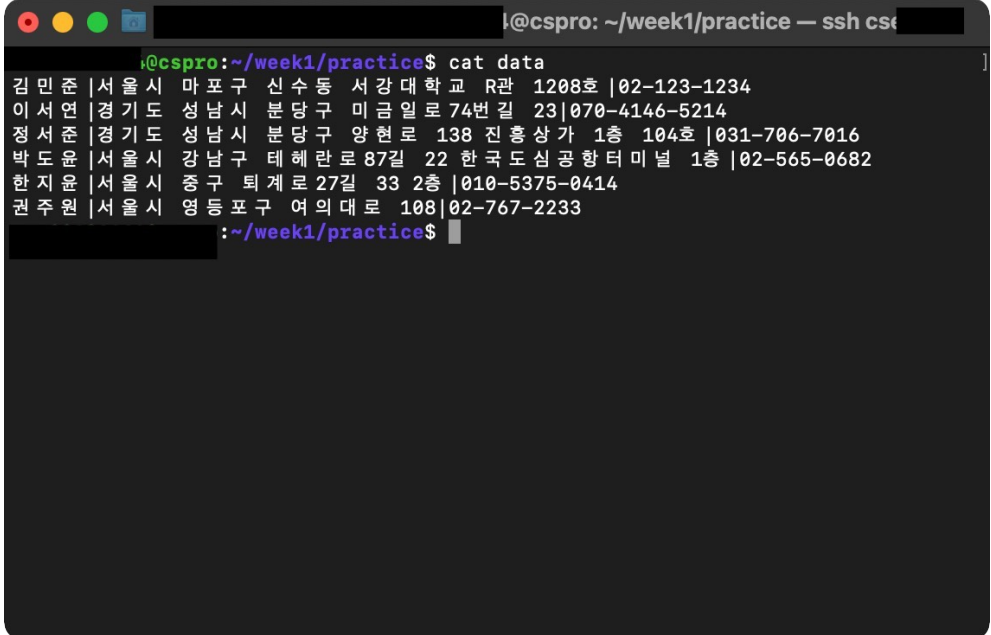
유닉스 시스템에 대해 미리 접해본 뒤 실험에 임할 수 있도록 한다. 아울러 부록에 있는 명령어에 대해 익숙해지도록 사용해본다.

2. 예비 학습

UNIX 시스템에 접속해본 뒤 자신의 홈 디렉토리를 확인해본다.

홈 디렉토리: /sogang/under/

셸 프로그래밍 실험에서 사용할 데이터 파일인 전화번호부를 만들어본다. 단 데이터 파일의 형식은 실험에서 나온 예제에 따르도록 한다. 5명 이상이 들어가 있는 데이터를 만들되 vi 에디터를 이용하여 작성한다. 단 파일명은 data로 한다.



```
@cspro: ~/week1/practice — ssh cspro@cspro
@cspro:~/week1/practice$ cat data
김민준|서울시 마포구 신수동 서강대학교 R관 1208호|02-123-1234
이서연|경기도 성남시 분당구 미금일로 74번길 23|070-4146-5214
정서준|경기도 성남시 분당구 양현로 138 진흥상가 1층 104호|031-706-7016
박도윤|서울시 강남구 테헤란로 87길 22 한국도심공항터미널 1층|02-565-0682
한지윤|서울시 중구 퇴계로 27길 33 2층|010-5375-0414
권주원|서울시 영등포구 여의대로 108|02-767-2233
@cspro:~/week1/practice$
```

위의 예제를 편집하는데 사용한 vi 명령어들을 나열하고, 해당 명령 수행하는 결과를 적어보도록 한다.

```
vi data: vi 편집기를 시작해 파일명이 "data"인 파일을 생성 및 편집한다.  
i: 입력 모드로 전환한 뒤 커서 위치 앞에서 삽입한다.  
esc: 입력 모드를 종료한다.  
wq: 변경사항을 보관한 뒤 vi를 종료한다.
```

위에서 작성한 데이터 파일을 \$home/.data 파일로 복사한다. 복사하기 위하여 사용한 명령들을 적어보도록 한다.

```
data를 작성한 디렉토리 안에서 다음과 같이 입력했다.  
cp ./data ~/
```

\$home/.data 파일을 그룹 및 다른 사용자가 아무 권한도 갖지 않도록 권한 변경을 해본다. 사용한 명령을 적어보도록 한다.

```
chmod 700 data
```

디렉토리에 대한 읽기, 쓰기, 실행 권한을 설정해보고 각각이 갖는 의미를 살펴본다.

```
chmod -R 754 week1
```

chmod는 파일의 접근 권한을 변경하기 위해 사용되는 명령어이다. -R은 변경사항을 하위 디렉토리와 파일에도 적용하기 위해 사용된 옵션이다. 754는 모드에 해당하는데, 소유자, 그룹, 다른 사용자 순으로 부여할 권한을 적어준 것이다. 읽기 권한은 4, 쓰기 권한은 2, 실행 권한은 1이며 부여하고자 하는 권한의 숫자를 더해 각 자리에 적어준다. 즉 7은 읽기 권한(4), 쓰기 권한(2), 실행 권한(1)을 모두 부여한 것이고, 5는 읽기 권한(4), 실행 권한(1)을 부여한 것이며, 4는 읽기 권한(4)만을 부여한 것이다. week1은 권한을 설정하고자 하는 디렉토리 이름이다.

3. 보충 학습

Regular Expression에 대하여 정리해보도록 한다.

실험에서 문자열을 검색하기 위해 egrep을 사용하는 쉘 프로그램을 작성한다. egrep은 grep을 확장한 형태이며 regular expression(정규 표현식)을 제공하는 grep이다. 여기서 정규 표현식이란 정규식이라고도 하며, 특정 규칙을 가지는 문자열의 집합을 표현할 때 사용되는 형식 언어이다. 다양한 텍스트 및 문자열 처리에 유용하게 사용된다.

메타문자(metacharacter)는 regular expression에서 사용되는 기호로, 내부적으로 특정한 의미를 가지는 문자이다. 다음과 같은 메타 문자들이 있다.

구문	설명	예시
.	null을 제외한 임의의 문자와 일치	a.b → abb, acb 일치, acc 불일치
+	한 번 이상 발생 수 일치	a+ → a. aaa 일치, bbb 불일치
?	0 or 1번 발생 수 일치	ab?c → abc, ac 일치, abbc 불일치
*	0번 이상 발생 수 일치	ab*c → ac,abc,abbc 일치, abb 불일치
{m}	m번 발생 수 일치	a{3} → aaa 일치, aa 불일치
{m, }	최소 m번 이상 발생 수 일치	a{3, } → aaa, aaaa일치, aa 불일치
{m, n}	최소 m번 이상, 최대 n번 이하 발생 수 일치	a{3,5} → aaa. aaaaa일치, aa 불일치
[...]	괄호 내 리스트에 있는 문자와 일치	[abc] → all, bill, cold 일치, doll 불일치
	or, 여러 대안 중 하나와 일치	a b → a 또는 b와 일치
(...)	괄호 내 표현식을 한 단위로 취급	(abc)?def → abcdefg, def 일치, ghi 불일치
^	문자열 시작 부분과 일치	^def → defghi 일치, abcdef 불일치
\$	문자열 끝 부분과 일치	def\$ → abcdef 일치, defghi 불일치
\	뒤에 나오는 메타문자를 일반 문자로 인식	\+ → abc+def 일치, abcdef 불일치
\n	괄호안에 그룹화된 n번째 식과 일치	(abc(def)xy)\1 → abcxycabc, defxydef 일치, abcxycdef. abcxyc 불일치
\d	숫자문자	\d{3} → 123, 345 일치, 1231 불일치
[[:class:]]	지정된 문자 클래스에 속한 임의의 문자와 일치	[[:upper:]]+ → 하나 이상의 연속되는 대문자 검색, abcDEFghi 일치, abcdefghi 불일치
[^:class:]	괄호안에 리스트에 없는 임의의 단일 문자와 일치	[^abc] → abcdef 일치, a,b,c 불일치