

1주차 예비보고서

전공: 신문방송학과 학년: 3학년 학번: 20191150 이름: 전현길

2.

1) 홈 디렉토리: /sogang/under/cse20191150

2) (데이터 파일)

김동현|서울시 마포구 신수동 서강대학교 AS관 301호|02-123-4567

박수연|서울시 마포구 노고산동 107-36번지|02-234-5678

김광석|서울시 서대문구 연희동 188-37번지|02-345-6789

정하늘|서울시 관악구 봉천동 635-503번지|02-456-7890

백지현|서울시 동작구 신대방동 360-17번지|02-567-8901

3)

vi data	'data'라는 이름의 파일을 생성 및 편집
i	현재 커서 자리에 입력, 또는 명령 모드에서 입력 모드로 전환
ESC + :	콜론 모드로 전환
wq	저장 후 종료
yy	현재 라인 복사
p	현재 커서 다음 줄에 붙여넣기

4) cse20191150@cspro:~\$ cp data .data

경로를 따로 지정하지 않을 경우, 현재 폴더(홈 디렉토리)에 생성된다.

5) cse20191150@cspro:~\$ chmod 700 .data

6) 파일 권한을 지정하는 명령어는 'chmod [3자리 8진수 숫자] [파일명]'이다. 8진수 숫자의 각 자리는 사용자, 그룹, 다른 사용자가 가진 권한을 의미하며, 각 8진수 숫자는 권한의 내용을 설명한다.

권한의 내용은 구체적으로 **읽기 권한**, **쓰기 권한**, **실행 권한**으로 구분된다. 읽기 권한이 있을 경우 파일의 내용을 읽을 수 있고, 쓰기 권한이 있을 경우 파일을 수정할 수 있으며, 실행 권한이 있을 경우 파일을 실행할 수 있다. **파일을 삭제하기 위해서는 쓰기 권한, 실행 권한이 둘 다 필요하다.**

파일 권한의 존재 여부는 읽기 권한, 쓰기 권한, 실행 권한의 값을 모두 합한 값으

로 표현된다. ex) `rwh` | $4 + 2 + 1 = 7$, `-wh` | $0 + 2 + 1 = 3$

3.

정규 표현식(Regular Expression)이란 특정한 양의 문자를 설명하는 패턴¹⁾이거나, 이러한 패턴을 표현할 때 사용하는 형식을 의미한다. 정규 표현식은 복잡한 문자열에 대한 검색·수정·삭제 작업을 효율적으로 수행하기 위해 고안되었으며, 데이터 분석이나 딥 러닝, 검색 엔진, 워드 프로세서 등 문자열 처리와 관련된 다양한 작업에서 응용된다.

정규 표현식의 기반 개념은 수학자 스티븐 클레이니(Stephen Kleene)의 정규 집합(regular set)에서 유래되었으며, 이후 1992년 Posix.2에서 표준 형태가 제시되어 다양한 프로그램에 응용되게 되었다. 웹사이트 제작에 특화된 범용 프로그래밍 언어 PHP(Hypertext Preprocessor)의 경우 대표적으로 POSIX와 PCRE라는 두 가지 정규 표현식을 지원하고 있다.

정규 표현식은 크게 **패턴(pattern)**과 **플래그(flag)**라는 두 영역으로 구분된다. 패턴은 두 `'/'` 기호 사이에서 표현되며, 패턴의 뒤에 플래그(flag)가 작성된다. 플래그는 검색 설정에 대한 옵션을 제공한다.

플래그	기능
i	대소문자를 구분하지 않음
g	대상 문자열과 일치하는 모든 문자를 선택함
m	여러 줄의 입력 문자열에 대해 검색함
y	대상 문자열의 현재 위치부터 비교를 시작함
p	대상 문자열이 UTF-8 인코딩된 것으로 설정함

패턴을 나타내기 위해서 **구분 문자(separate character)**, **메타 문자(meta character)**, **리터럴(literal)**이라는 세 가지 형태의 문자가 사용된다. **구분 문자**는 정규식과 일반 문자열을 구분하기 위한 문자를 의미하며, 앞선 `'/'` 기호 등이 이에 해당된다. **메타 문자**는 정규 표현식에서 특정한 의미를 갖고 사용되는 특수 문자를 의미한다. 마지막으로, **리터럴**은 정규 표현식에서 '문자 그대로' 사용되며 다른 의미가 없는 문자를 의미한다. 아래의 표는 메타 문자의 예이다.

메타 문자	기능
.	\n을 제외한 모든 글자
()	패턴 요소들을 하나의 요소로 묶음

1) regular expressions tutorial. <https://www.regular-expressions.info/tutorial.html>.

*	앞의 문자가 없거나, 2개 이상 존재, ex) a*
?	앞의 문자가 없거나, 1개 존재
+	앞의 문자가 1개 이상 존재
{m, n}	앞의 문자가 m개 이상 n개 이하 존재, ex) a{m, n}
[...]	가능한 문자열의 집합 검색, ex) [aeiou]
[a b]	a 또는 b에 해당될 경우 검색, ex [hi hello]

정규 표현식의 문법들은 크게 문자 클래스(character classes), 앵커(anchors), 이스케이프 문자(escaped characters), 그룹(groups), 전후방탐색자(lookaround), 수량자(quantifiers), 선택자(alternation)로 구분된다.²⁾

2) 배워보자 정규표현식! (Regular Expression), <https://wooncloud.tistory.com/113>.