```
import re
```

정규표현식, 문자열에서 패턴찾기

```
text = 'My id number is [G203_5A]'
text
```

```
'My id number is [G203_5A]'
```

```
result = re.findall('a',text)   # re.findall(찾을 문자, 찾아낼 데이터)
result
```

```
[]
```

```
result = re.findall('A',text)
result
```

```
['A']
```

```
result = re.findall('i',text)
result
```

```
['i', 'i']
```

```
# 소문자 연속해서 찾기(단어가 연속되어 있으면 묶어줌) '[a-z]+'
result = re.findall('[a-z]+',text)
result
```

```
['y', 'id', 'number', 'is']
```

```
# 소문자 연속해서 찾기2 '[a-z]'
result = re.findall('[a-z]',text)
result
```

```
['y', 'i', 'd', 'n', 'u', 'm', 'b', 'e', 'r', 'i', 's']
```

```
# 대문자 연속해서 찾기
result = re.findall('[A-Z]+',text)
result
```

```
['M', 'G', 'A']
```

```
# 숫자 찾기
result = re.findall('[0-9]',text)
result
```

```
['2', '0', '3', '5']
```

```
# 숫자 찾기2 (붙은 건 붙은 그대로)
result = re.findall('[0-9]+',text)
result
```

```
['203', '5']
```

```
# 영문자 및 숫자 찾기
result = re.findall('[a-zA-Z0-9]',text)
result
```

```
['M',
 'y',
 'i',
 'd',
 'n',
 'u',
 'm',
 'b',
 'e',
 'r',
 'i',
 's',
 'G',
 '2',
 '0',
 '3',
 '5',
 'A']
```

```
# 영문자 및 숫자 연속해서 찾기
result = re.findall('[a-zA-Z0-9]+', text)
result
```

```
['My', 'id', 'number', 'is', 'G203', '5A']
```

```
# 영문자 및 숫자가 '아닌것' 찾기
result = re.findall('[^a-zA-Z0-9]',text)
result
```

```
[' ', ' ', ' ', ' ', '[', '_', ']']
```

```
# 영문자 및 '_' 특수기호 찾기
result = re.findall('[₩w]',text)
result
```

```
['M',
 'y',
 'i',
 'd',
 'n',
 'u',
 'm',
 'b',
 'e',
 'r',
 'i',
 's',
 'G',
 '2',
 '0',
 '3',
 '_',
 '5',
 'A']
```

```
# 영문자 및 '_' 특수기호 연속해서 찾기
result = re.findall('[₩w]+',text)
result
```

```
['My', 'id', 'number', 'is', 'G203_5A']
```

```
# 영문자 및 '_'특수기호 아닌 문자 찾기
result = re.findall('[₩W]', text)
result
```

```
[' ', ' ', ' ', ' ', '[', ']']
```

문자열에서 특정이름 찾아내기

```
# ₩w ( 1 char )    ##한 글자
# ₩d ( 1 decimal )
# ₩s ( 1 space )

# + ( 1, ..., N )
# ? ( 0, 1 )
# * ( 0, 1, .. N )

# ₩d{N} ( 숫자가 N개 나온다. )   ##숫자가 3자리인 것
# ₩d{N,M} ( 숫자가 N~M개 나온다 )  ## 숫자가 N~M자리인 것
```

```
text = """
옛날 옛적에 김길동라는 사람이 살았습니다.
그에게는 5형제가 있었는데, 김찬동, 김찬식, 김찬이, 김찬혁, 김찬민 이렇게 5명 있었습니다.
그리고 그는 결혼을 해서 김수용, 김지용, 김무용 3남매를 낳고 행복하게 잘 살았습니다.
"""
```

```
pattern = re.compile('김찬₩w')
pattern
```

```
re.compile(r'김찬₩w', re.UNICODE)
```

```
brother = pattern.findall(text)
brother
```

```
['김찬동', '김찬식', '김찬이', '김찬혁', '김찬민']
```

```
pattern = re.compile('김.용')
#pattern = re.compile('김₩w영')
#둘은 동일
```

```
children = pattern.findall(text)
children
```

```
['김수용', '김지용', '김무용']
```

```
brother = set(brother)
brother
```

```
{'김찬동', '김찬민', '김찬식', '김찬이', '김찬혁'}
```

```
brother = sorted(brother)
brother
```

```
['김찬동', '김찬민', '김찬식', '김찬이', '김찬혁']
```

```
text = 'A sky, a dragonfly aand a butterfly!'
```

```
pattern = re.compile('₩w+fly')
pattern.findall(text)
```

```
['dragonfly', 'butterfly']
```

핸드폰 번호에 대한 파싱

> \d{2,3}[-.]?\d{3,4}[-.]?\d{4}

```
text = """
    010-5670-3847    # space, -, . => []
    010 5670 3847
    010.5670 3847
"""
```

```
pattern = re.compile('₩d{3}[ -₩.]?₩d{4}[ -₩.]?₩d{4}')  #.이나 공백이나 -
```

```
pattern.findall(text)
```

```
['010-5670-3847', '010 5670 3847', '010.5670 3847']
```

```
text = """
    010-5670-3847    # space, -, . => []
    옛날에는 011-1052-3847 이랬는데..
    010 5670 3847
    010.-5670 3847
    사는동네가 금영아파트 103동112호
    그리구, 사무실번호는 02-3450-4012이고
    우편번호는 100-711, 청파로 887번지

"""
```

```
pattern = re.compile("₩d{2,3}[ -₩.]{1,2}?₩d{3,4}[ -₩.]?₩d{4}")
# ? : 숫자가 {3}거나 {3,4} 이거나 {4}이거나
```

```
pattern.findall(text)
```

```
['010-5670-3847',
 '011-1052-3847',
 '010 5670 3847',
 '010.-5670 3847',
 '02-3450-4012']
```

주민번호에 대한 파싱

> - 뒷자리를 암호화 시키기
> - 숫자 6자리 - 숫자 7자리

```
text = """
    김부영 990811-3097532
    김오영 011108-3123441
    김지영 111009-4085122
"""
```

```
pattern = re.compile("₩d{6}-?₩d{7}")    # ?₩d
                                        # -가 있을 수도 있고...붙을 수도 있고...
```

```
pattern.findall(text)
```

```
['990811-3097532', '011108-3123441', '111009-4085122']
```

```
pattern = re.compile("₩d{6}-₩d{7}")
# 정규표현식 GROUP
# 1. 생년월일 그룹 <birth>
# 2. 주민등록번호 뒷자리 그룹 <secret>
```

```
pattern = re.compile("(?P<birth>₩d{6})-(?P<secret>₩d{7})")
# ?P<>로 패턴 파라미터
```

```
pattern.findall(text)
```

```
[('990811', '3097532'), ('011108', '3123441'), ('111009', '4085122')]
```

```
# ₩g로 불러오기
result = pattern.sub("₩g<birth>-*******",text)
print(result)
```

```

        김부영 990811-*******
        김오영 011108-*******
        김지영 111009-*******

```

```
pattern = re.compile('(?P<name>₩w{3}) (?P<birth>₩d{6})-(?P<secret>₩d{7})')
```

```
pattern.findall(text)
```

```
[('김부영', '990811', '3097532'),
 ('김오영', '011108', '3123441'),
 ('김지영', '111009', '4085122')]
```

```
print(text)
```

```

        김부영 990811-3097532
        김오영 011108-3123441
        김지영 111009-4085122

```

```
# 암호화
result = pattern.sub("₩g<name>(₩g<birth>-*******)",text)
result
```

```
'₩n    김부영(990811-*******)₩n    김오영(011108-*******)₩n    김지영(111009-*******)₩n'
```

```
result = result.split('₩n')
result
```

```
['',
 '    김부영(990811-*******)',
 '    김오영(011108-*******)',
 '    김지영(111009-*******)',
 '']
```

```
result.pop(0)
result
```

```
['    김부영(990811-*******)',
 '    김오영(011108-*******)',
 '    김지영(111009-*******)',
 '']
```

```
result.pop(-1)
result
```

```
['    김부영(990811-*******)',
 '    김오영(011108-*******)',
 '    김지영(111009-*******)']
```

```
for idx, val in enumerate(result) :
    val = val.replace(" ", "")
    print(idx,val)
```

```
0 김부영(990811-*******)
1 김오영(011108-*******)
2 김지영(111009-*******)
```

Quiz

- Email 패턴 : ID + @ + URL

```
# 작업 폴더 확인
%ls data
```

```
with open('./data/emails.txt', encoding ='utf-8') as fp :
    data = fp.read()
    fp.close()
```

```
data
```

```
'Id₩tDocNumber₩tMetadataSubject₩tMetadataTo₩tMetadataFrom₩tSenderPersonId₩tMetadataDateSent
```

```
pattern = re.compile("₩w[a-zA-Z0-9]{0,12}[ -₩.]?₩w@.?₩w{0,12}.?₩w{0,12}")
```

```
result = pattern.findall(data)
```

```
result
```

```
['Sullivan11@state.gov',
 'Sullivan11@state.gov',
 '_Brose@armed-servic',
 'MillsCD@state.gov',
 'MillsCD@state.gov',
 'lhdr22@clintonernail.com',
 'hdr22@clintonemail corn',
 'MillsCD@state.gov',
 'MillsCD@state.gov',
 'snipead@state.gov',
 'hrod17@clintonemail.com',
 'hrod17@clintonemail.com',
 'Russorv@state.gov',
 'hrod17@clintonernailcom',
 '짜Russorv@state.gov',
 'hrod17@clintonernailcom',
 '짜Russorv@state.gov',
 'MillsCD@state.gov',
 'MillsCD@state.gov',
 'hrod17@clintonemail.corn',
 'hrod17@clintonemail.corn',
 'Sullivanli@stategov>',
 'Sullivanli@stategov>',
 'SullivanJJ@state.gov',
 'SullivanJJ@state.gov',
 'Reinesp@stategov;',
 'Reinesp@stategov;',
 'MillsCD@state.gov',
 'MillsCD@state.gov',
 'snipead@state,gov',
 'KohliFf@state.gov',
 'KohliFf@state.gov',
 'ShermanWR@state.gov',
 'ShermanWR@state.gov',
 'ShermanWR@state.gov',
 'Sullivanii@state.gov',
 'Sullivanii@state.gov',
 'Sullivanii@state,gov',
 'Sullivanii@state,gov',
 'idr22@dintonemail.com',
 'sullivanjj@state.gov',
 'hrod17@clintonernaii.com',
 'esullivanjj@state.gov',
 'hrod17@clintonernaii.com',
 'esullivanjj@state.gov',
 'Russorv@state.gov',
 'hrod17@clintoriernail',
 '짜Russont@state.gov짜',
 'hrod17@clintoriernail',
 '짜Russont@state.gov짜',
 'Russorv@state.gov',
 "Russorv@stategov'",
 "Russorv@stategov'",
 'sullivanjj@state.gov',
 'sullivanjj@state.gov',
 'hrod17@clintonemail.com',
 'hrod17@clintonemail.com',
 'sullivanii@state.gov',
sullivanjj@state.gov',
'특ullivann@state.gov',
'특ullivann@state.gov',
'HDR22@clintonemail.corn',
'HDR22@clintonernallycorn',
'SullivanD@state,gov',
'Sullivanit@state,gov',
'Sullivanit@state,gov',
'HDR22@clintonemall,corn',
'MillsCD@stategov>',
'MillsCD@stategov>',
'AbedinH@state.goy',
'AbedinH@state.goy',
'BitterR@state,gov',
'SullivanJJ@state.gov',
'MillsCD@state.gov',
'MillsCD@state.gov',
'Russorv@state.gov',
'Russorv@state.gov쨔',
'hrod17@clintonemail.com',
'SullivanJJ@state.gov',
'hrod17@clintonemail.com',
'Russorv@state.gov쨔',
'SullivanJJ@state.gov',
'Sullivarill@state.gov',
'Sullivarill@state.gov',
'MillsCD@state.gov',
'MillsCD@state.gov',
'MillsCD@state.gov',
'MillsCD@state.gov',
'AbedinH@state.gov',
'AbedinH@state.gov',
'MillsCD@state.gov',
'MarshallCP@stategov>',
'HOR22@clintonemail.com',
'hrod17@clintonemail.com',
'hrod17@clintonemail.com',
'Sullivan13@stategov]',
'sullivanjj@state.gov',
'sullivanjj@state.gov',
'hrod17@dintonemailcom',
'hrod17@dintonemailcom',
'sullivanjj@state.gov',
'11@state.govi',
'MillsCD@state.gov',
'MillsCD@state.gov',
'sullivanjj@state.gov',
'sullivanjj@state.gov',
'hrod17@clintonemail.com',
'hrod17@clintonemail.com',
'sullivanjj@state.gov',
'SullivanAt@state.gov',
'sullivanjj@state.gov',
'gll@state.gov',
'hrod17@clintonemail.com',
'hrod17@clintonemail.com',
'gll@state.gov',
'MarshallCP@state.gov',
'MarshallCP@state.gov',
```