REGULAR EXPRESSION

```
import re

str="its a rainy day"

res=re.findall("\Aits",str)
print(res)

if res:
  print("found")
else:
  print("Not found")
```

```
['its']
found
```

```
res1=re.search(r'nice','hi i am a human \ i am not nice')
print(res1)
print(res1.group())
print("start",res1.start())
print("end ",res1.end())

match = re.search(r'portal', 'A computer science \ portal forEducation')
print(match)
print(match.group())
print('Start Index:', match.start())
print('End Index:', match.end())
```

```
<re.Match object; span=(27, 31), match='nice'>
nice
start 27
end  31
<re.Match object; span=(21, 27), match='portal'>
portal
Start Index: 21
End Index: 27
```

```
print(re.findall(r'[E-m]ducation', 'Education of education: \ A computer science portal for education mducation'))
```

```
['Education', 'education', 'education', 'mducation']
```

```
print('Range',re.search(r'[a-zA-Z]', 'x'))
```

```
Range <re.Match object; span=(0, 1), match='x'>
```

```
x = range(3, 6)
for n in x:
 print(n)

print("printed in the difference of 2")
x = range(3, 20, 2)
for n in x:
 print(n)
```

```
3
4
5
printed in the difference of 2
3
5
7
9
11
13
15
17
19
```

```
print(re.search(r'[^a-z]', 'c'))


print(re.search(r'C[^l]', 'Class'))
```

```
None
None
```

```
# Beginning of String
match = re.search(r'^is', 'This is the month')
print('Beg. of String:', match)
match = re.search(r'^is', 'is the month')
print('Beg. of String:', match)
# End of String
match = re.search(r'education$','Compute science portal for education')
print('End of String:', match)
```

```
Beg. of String: None
Beg. of String: <re.Match object; span=(0, 2), match='is'>
End of String: <re.Match object; span=(27, 36), match='education'>
```

```
print('Any Character', re.search(r'p.th.n', 'python 3'))
```

```
Any Character <re.Match object; span=(0, 6), match='python'>
```

```
print('Color',re.search(r'colou?r', 'color'))
print('Colour',re.search(r'colou?r', 'colour'))
```

```
Color <re.Match object; span=(0, 5), match='color'>
Colour <re.Match object; span=(0, 6), match='colour'>
```

```
print('Date{mm-dd-yyyy}:', re.search(r'[\d]{2}-[\d]{2}-[\d]{4}','13-07-2023'))
```

```
Date{mm-dd-yyyy}: <re.Match object; span=(0, 10), match='13-07-2023'>
```

```
print('Three Digit:', re.search(r'[\d]{3,4}', '189'))
print('Four Digit:', re.search(r'[\d]{3,4}', '2145'))
```

```
Three Digit: <re.Match object; span=(0, 3), match='189'>
Four Digit: <re.Match object; span=(0, 4), match='2145'>
```

```
print(re.search(r'[\d]{1,}','5th Floor, B-218,\Sector-136, Noida, Uttar Pradesh - 201405'))
```

```
<re.Match object; span=(0, 1), match='5'>
```

```
print(re.search(r'[\d]+','5th Floor, B-218,\Sector-136, Noida, Uttar Pradesh - 201405'))
```

```
<re.Match object; span=(0, 1), match='5'>
```

```
grp = re.search(r'([\d]{2})-([\d]{2})-([\d]{4})', '12-07-2023')
print(grp)
```

```
<re.Match object; span=(0, 10), match='12-07-2023'>
```

```
grp = re.search(r'([\d]{2})-([\d]{2})-([\d]{4})','14-07-2023')
print(grp.groups())
```

```
('14', '07', '2023')
```

```
grp = re.search(r'([\d]{2})-([\d]{2})-([\d]{4})','14-07-2023')
print(grp.group(3))
```

```
2023
```

```
grp = re.search(r'(?P<dd>[\d]{2})-(?P<mm>[\d]{2})-(?P<yyyy>[\d]{4})','14-07-2023')
print(grp.group('dd'))

grp = re.search(r'(?P<dd>[\d]{2})-(?P<mm>[\d]{2})-(?P<yyyy>[\d]{4})','14-07-2023')
print(grp.groupdict())
```

```
14
{'dd': '14', 'mm': '07', 'yyyy': '2023'}
```

```
print('negation:', re.search(r'n[^e]', 'Python'))
print('lookahead:', re.search(r'n(?!e)', 'Python'))
```

```
negation: None
lookahead: <re.Match object; span=(5, 6), match='n'>
```

```
print('positive lookahead', re.search(r'n(?=e)', 'jasmine'))
```

```
positive lookahead <re.Match object; span=(5, 6), match='n'>
```

```
print(re.sub(r'([\d]{4})-([\d]{4})-([\d]{4})-([\d]{4})',r'\1\2\3\4', '1111-2222-3333-4444'))
```

```
1111222233334444
```

```
regex = re.compile(r'([\d]{2})-([\d]{2})-([\d]{4})')
```

```
print('compiled reg expr', regex.search('13-07-2023'))

print(regex.sub(r'\1.\2.\3', '13-07-2023'))
```

```
compiled reg expr <re.Match object; span=(0, 10), match='13-07-2023'>
13.07.2023
```

✓ 0s completed at 8:57 PM