re.IGNORECASE, re.DOTALL, and re.VERBOSE

re.IGNORECASE

- Matches text without considering uppercase/lowercase.

Example:

- re.search(r"hello", "HeLLo World", re.IGNORECASE)

Example

```
import re

text = "HeLLo Students"

pattern = r"hello"

match = re.search(pattern, text, re.IGNORECASE)

if match:
    print("Match found:", match.group())
else:
    print("Not found")
```

re.DOTALL

- Makes dot (.) match newline characters also.

```
import re

text = """Hello everyone,
Welcome to Regex class."""

pattern = r"Hello.*class"

match = re.search(pattern, text, re.DOTALL)

if match:
    print("Match found across lines:")
    print(match.group())
else:
    print("Not found")
```

re.VERBOSE

- Allows multi-line, readable regex with comments.

Example:

```
import re

pattern = re.compile(r"""
    ^           # start of string
    \d{3}       # first 3 digits
    -           # hyphen
    \d{2}       # next 2 digits
    -           # hyphen
    \d{4}       # last 4 digits
    $           # end of string
""", re.VERBOSE)

text = "123-45-6789"

if pattern.match(text):
    print("Valid format")
else:
    print("Invalid format")
```

Combining All Flags

- We use bitwise OR (|) to combine flags.

Example:

- flags = re.IGNORECASE | re.DOTALL | re.VERBOSE

Pretty Printing in Python

Pretty Printing (or **pprint**) is a technique to display data structures—like lists, dictionaries, nested structures—in a readable and well-formatted manner.

Python provides a built-in module:

import pprint, pformat

Example

```
import pprint
message = 'It was a bright cold day in April, and the clocks were striking thirteen.'
count = {}
for character in message:
    count.setdefault(character, 0)
    count[character] = count[character] + 1
pprint.pprint(count)
```

use Pretty Print

- To make large or nested data structures readable
- Useful for debugging
- Makes output clean and professional

pformat() in Python

- pformat() stands for **Pretty Format**.
  It **formats** complex Python data structures into a **neatly aligned string** but does **NOT print** it automatically.
- It is part of the **pprint module**:

from pprint import pformat

use pformat()

- Use pformat() when you want:
  - ✓ a neatly formatted string
  - ✓ to store pretty output in a variable
  - ✓ to write pretty data to a log file
  - ✓ to print it later inside another print statement
  - ✓ to display readable dictionary/list output in GUIs, logs, files

Example

```
from pprint import pformat

data = {
    "name": "Soumya",
    "subjects": ["Python", "AI", "ML"],
    "marks": {"python": 98, "ai": 92, "ml": 90},
    "location": {"city": "Bengaluru", "pincode": 560001}
}

result = pformat(data)
print(result)
```

```
from pprint import pformat
data = {"numbers": list(range(20))}
print(pformat(data, width=30))
```

Output:?