Regulære Uttrykk

En sekvens av tegn som spesifiserer et mønster i tekst.

Et regulært uttrykk definerer hva vi søker etter; en regel. Det kan brukes til å hente ut det man ønsker etter i en tekst.

Tegn

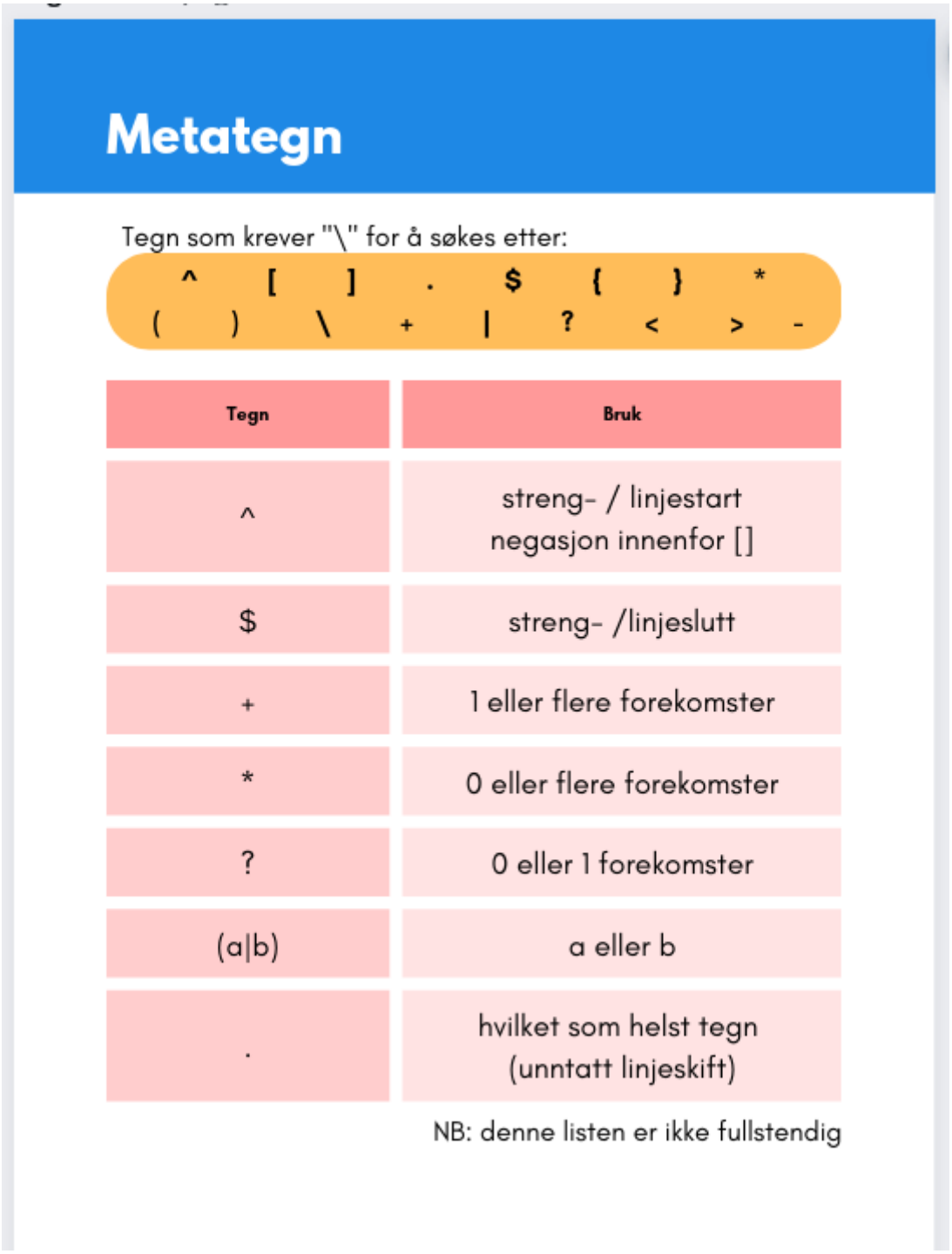
eng. *characters*

Bokstaver

- vanlig tolkning av bokstaver. (f.eks. RU /a/ refererer til bokstaven "a" i en tekst)

Metategn

- spesialtegn som lar oss sette uttrykk sammen.



Regulære Uttrykk i Python

Methoder

- `re.findall(mønstret, string)`
  - returnerer en liste med alle treff
- `re.split(mønstret, string)`
  - returnerer en liste som inneholder substrengene i strengen delt opp på hvert treff
- `re.sub(mønstret, ny, string)`
  - erstatter treff av mønstret med den nye verdien i strengen

re.findall()

Finne alle siffre i strengen: `"6 januar 1992"`

In [ ]:
```
import re    # import regex modulen

tekst = "6 januar 1992"

uttrykk = r"\d+"    # \d finner sifre og + spesifiserer 1 eller flere for

tokens = re.findall(uttrykk, tekst)

print("Dato:", tekst)
print("Tall:", tokens)
```

```
Dato: 6 januar 1992
Tall: ['6', '1992']
```

re.split()

Finne rotene til de følgende verbene: `"spiser danser skriver"`

In [ ]:
```
import re    # import regex modulen

tekst = "spiser danser skriver"

uttrykk = r"er\s?"    # Deler på "-er" med optional mellomrom

tokens = re.split(uttrykk, tekst)

print("Dato:", tekst)
print("Tall:", tokens)
```

```
Dato: spiser danser skriver
Tall: ['spis', 'dans', 'skriv', '']
```

`re.sub()`

Eksempel: Vi jobber med data om kunder i en bedrift og vil fjerne personopplysningene fra filen.

Dataen inneholder:

- en kundeID som består av "K" følgt av 5 bokstaver/sifer
- en kontonummer som består av 6 sifre
- en fødselsdato

og er organisert slik:

```
"""
KundeID: K-----
Kontonummer: ######
Fødselsdato: ##/##/####
```

Erstatt personopplysningene i følgende strengen med `"#"`:

```
"""
KundeID: K34ABC
Kontonummer: 123456
Fødselsdag: 01/23/1992
"""
```

Ressurser

- Regex101
- Regex Cheat Sheet (På gruppesiden eller bare google det)
- W3 Schools Tutorial