Uke 9 - konsept

Gruppe 14

Kristin - kristlgj@ifi.uio.no

Menti

Plan for gruppetimen

- Recap av pensum og oppgaver
  - Magiske metoder
    - __eq__ og sammenligning av objekter
    - __str__
  - Større oppgave med samlinger av objekter

Hva ble gjennomgått på forelesningen?

Tenk selv, så diskuter

- Hva lærte du, hva skjønte du ikke?
- Se gjerne over slides fra forelesningen om dere ikke husker!
- Er det noe dere lurer på?

Menti

Magiske metoder

- Bestemte metoder som man aldri eksplisitt kaller på (kaller aldri på metodenavnene)
- Skrives med dobbel understrek både på starten og slutten av metodenavnet
- F.eks. __init__ som dere allerede har brukt!

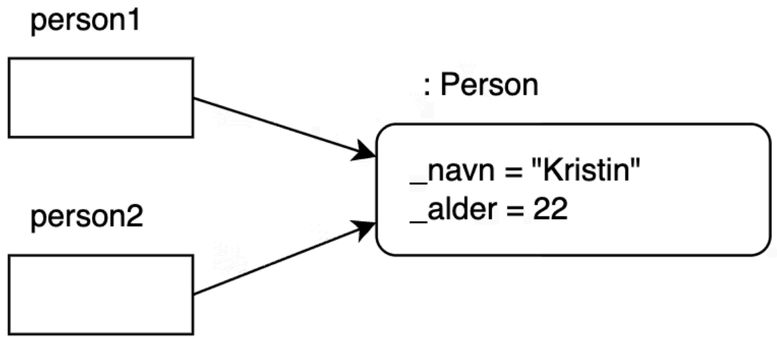
- __str__
- __eq__

Man kan sammenligne:

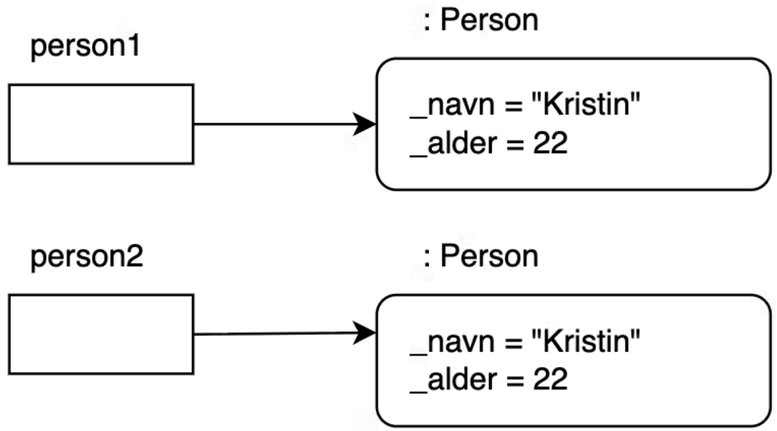
- Referanse
  - Er det samme objekt?
  - person1 is person2

- Objekt
  - Er det like objekter (objekter med like innhold)?
  - person1 == person2

Referanse (samme objekt)



Objekt (likt objekt)



__eq__ metode

- Tar et annet objekt som vi skal sammenligne med som parameter
- Returnerer True eller False
- Man kaller på denne metoden ved å bruke == (eller in)
- Uten __eq__ metode så sammenlignes kun referansene (om det er samme objekt)

```
class Person:
    def __init__(self, navn, alder):
        self._navn = navn
        self._alder = alder

    def __eq__(self, annen):
        return ((self._navn == annen._navn) and (self._alder == annen._alder))


def hovedprogram():
    person1 = Person("Kristin", 22)
    person2 = Person("Kristin", 22)

    person1 == person2    # Blir True

hovedprogram()
```

__str__ metode

- For å lage en streng-representasjon av et objekt (f.eks. for printing)
- Uten __str__ vil vi få utskrift av selve objektet på adressen
- Returnerer strengen vi vil skal representere objektet
- Man kaller på denne metoden ved å bruke print() eller str()

```python
class Person:
    def __init__(self, navn, alder):
        self._navn = navn
        self._alder = alder

    def __str__(self):
        return f"Navn: {self._navn}, Alder: {self._alder}"
```

Uten __**str**__ metode

<__main__.Person object at 0x105261a90>

Med __**str**__ metode

Navn: Kristin, Alder: 22

Oppgave 1

1. Hva blir skrevet ut til terminalen ved kjøring av programmet over? Kom frem til et svar før dere kjører programmet.
2. Hva skjer hvis vi bruker print(self._fødselsnummer + " (" + self._navn + ")") i stedet for return self._fødselsnummer + " (" + self._navn + ")" ?
3. Hva blir skrevet ut til terminalen ved fjerning av **str**-metoden over? Kom frem til et svar før dere kjører programmet.

```
class Person:
    def __init__(self, fødselsnummer, navn):
        self._fødselsnummer = fødselsnummer
        self._navn = navn


    def __str__(self):
        return self._fødselsnummer + " (" + self._navn + ")"
```

Oppgave 2

1. Hva skrives ut til terminalen med print(adr1 == adr2) i programmet over? Kom frem til et svar før dere kjører programmet.
2. Hva skrives ut til terminalen med print(adr1 is adr2) i programmet over? Kom frem til et svar før dere kjører programmet.
3. Sammenlign svarene i a) og b). Hva oppdager dere, og hvorfor er det slik?
4. Hvorfor trenger vi ikke egne metoder for å hente ut *gatenavn,* bolignummer og *postnummer fra annen-objektet i* _eq__-metoden?

```
class Adresse:
    def __init__(self, gatenavn, bolignummer, postnummer):
        self._gatenavn = gatenavn
        self._bolignummer = bolignummer
        self._postnummer = postnummer


    def __eq__(self, annen):
        return (self._gatenavn == annen._gatenavn and
        self._bolignummer == annen._bolignummer and
        self._postnummer == annen._postnummer)


adr1 = Adresse("Pythongata", 1, 0000)
adr2 = Adresse("Pythongata", 1, 0000)
print(adr1 == adr2)
print(adr1 is adr2)
```

Menti

Universitetssystem: Institutt for informatikk

Menti

- Klassene: Student, Emne, Studieprogram, Institutt
- Institutt skal holde på studenter, emner og studieprogrammer
- Hovedprogram der bruker kan få oversikt over studenter, oversikt over emner, oversikt over studieprogrammer, registrere studenter, legge et emne til en student, legge en student til et studieprogram -- så mye de vil helt til de velger å avslutte

Hvilke instansvariabler trenger vi for hver klasse? Hvilke metoder?

Menti

Instansvariabler

- Student -- navn, brukernavn, liste med emner
- Emne -- emnekode, emnenavn, studiepoeng
- Studieprogram -- navn på studieprogram, liste med studenter
- Institutt -- liste med studenter, liste med emner, liste med studieprogrammer

Metoder

- Student -- legg_til_emne(), hent_brukernavn()
- Emne -- hent_emnekode()
- Studieprogram -- legg_til_student()
- Institutt -- hent_studenter(), hent_emner(), hent_studieprogrammer(), registrer_student()

Alle bør ha __str__ og __eq__ metoder

Menti

Skriv ned en ting som ble klarere og en ting du fortsatt sliter med:) Og tilbakemeldinger på gruppetimene!