

Python Session 19



Today

- Ce este POO
- Abstractizare
- Incapsularea
- Modularizarea
- Ierarhizarea
- Obiecte
- Clase



Ce POO?

- Programarea Orientată Obiect (POO) este o metodă de proiectare și implementare în care programele sunt reprezentate sub forma unor colecții de obiecte (clase) care interacționează între ele prin intermediul mesajelor. Limbaje de programare orientate obiect:
- Python
- C#
- Java
- C++

Concepte de baza in POO

- Principalele concepte care stau la baza POO sunt:
- Abstractizare
- Încapsularea
- Modularizarea
- Ierarhizarea

Abstractizare:

- Abstractizarea este procesul de grupare a datelor și metodelor de prelucrare specifice rezolvării unei probleme.
- Abstractizare: exprimă toate caracteristicile esențiale ale unui obiect care fac ca acesta să se distingă de alte obiecte; oferă o definiție precisă a granițelor conceptuale ale obiectelor din perspectiva unui privitor extern.

Abstractizare:

- Abstractizarea se concentrează asupra caracteristicilor esențiale ale unui obiect, în raport cu perspectiva unui observator.

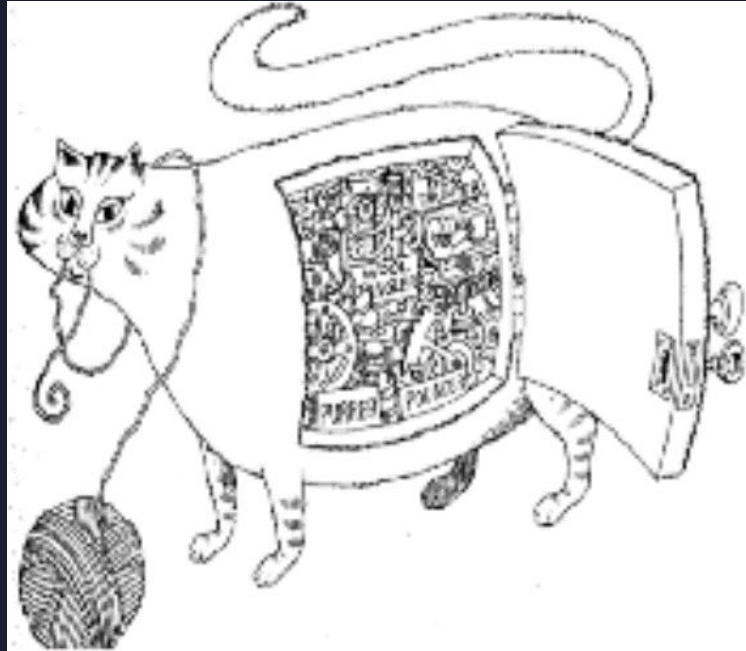


Încapsularea

- Gruparea datelor și metodelor aplicabile acestora într-o singură structură de date, definind totodată modul în care obiectul și restul programului pot referi datele din obiect.
- Concept care definește apartenența unor proprietăți și metode față de un obiect.
- Constă în separarea aspectelor externe ale unui obiect care sunt accesibile altor obiecte de aspectele interne ale obiectului care sunt ascunse celorlalte obiecte

Încapsularea

- **Încapsularea** ascunde detaliile implementării unui obiect



Modularizarea

- Modalitatea prin care un program este divizat în subunitati (module) ce pot fi compilate/rulate separate
- Un modul grupează abstracțiuni (clase) legate logic între ele



Ierarhizarea

Reprezintă o ordonare a abstractiunilor.

Principalele tipuri sunt:

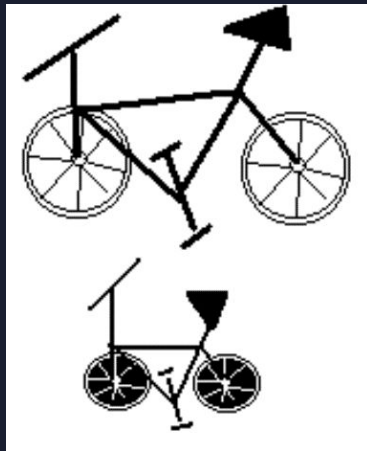
- **Moștenirea** (ierarhia de clase) relație între clase în care o clasă preia structura și comportamentul definit în una sau mai multe clase (semantic implică o relație de tip “*este un/o*”, eng. “*is a*”).
- **Agregarea** (ierarhia de obiecte) relație între două obiecte în care unul dintre obiecte aparține celui alt obiect. (semantic implică o relație de tip “*parte din*”, eng. “*part of*”)

Obiecte

- Un **obiect** este o reprezentare a unei entități din lumea reală asupra căruia se poate întreprinde o acțiune sau care poate întreprinde o acțiune
- Un obiect este caracterizat de:
 - *nume*
 - *attribute* (date) : valorile atributelor la un moment dat definesc o **stare metode** (servicii, operatii)

Clasa

- O **clasă** este o colecție de obiecte cu aceeași structură (caracteristici) și același comportament (metode sau operații)



Clasa Bicicletă

-Attribute:

tip cadru

dimensiunea rotii

numar de viteze

- Metode:

acelerează

frânează

Tipuri de date abstracte și obiecte

- A doua definiție pentru obiecte și clase:

O **clasa** este o implementare a unui tip de date abstract.

- Ea definește atributele și metodele care implementează structura de date respectiv operațiile tipului de date abstract.
- Un **obiect** este o *instanță* a unei clase. El este unic determinat de *numele* său și are o *stare* reprezentată de valorile atributelor sale la un anumit moment particular.