PROGRAMSKA POTPORA KOMUNIKACIJSKIM SUSTAVIMA

Prva laboratorijska vježba

Klasa *Professional*

Klasa *Professional* roditeljska je klasa koja služi kao temelj za stvaranje objekata koji predstavljaju osobe u različitim profesionalnim područjima poput umjetnika, znanstvenika i poduzetnika.

Atributi klase *Professional*:

* *first\_name* (*string*): ime osobe
* *last\_name* (*string*): prezime osobe
* *age* (*string*): dob osobe.
* *expertise\_level* (*int*): razina stručnosti osobe
* *salary* (*float*): plaća osobe
* *nationality* (*string*): nacionalnost osobe

Metode klase Professional:

* *\_\_init\_\_*(*self, first\_name, last\_name, age, expertise\_level, salary, nationality*): konstruktor klase *Professional*
* *name*(*self*) -> *str*: vraća puno ime osobe.
* *personal\_info*(*self*) -> *str*: vraća osnovne informacije o osobi (dob, nacionalnost, specijalnost)
* *achievements*(*self*) -> *str*: vraća postignuća osobe
* *calculate\_avg\_salary*(*professionals*: *List*[*Professional*]) -> *float*: statička metoda koja računa prosječnu plaću osobe
* *rank\_by\_salary*(*professionals*: *List*[*Professional*]) -> *List*[*Professional*]: statička metoda koja rangira (sortira) osobe silazno prema plaći
* *perform\_action*(*self*) -> *str*: vraća radnju koju osoba izvodi

Klasa *Artist*

Klasa *Artist* nasljeđuje klasu *Professional* i predstavlja osobu koja je bavi umjetnošću.

Atributi klase *Artist*:

* svi atributi iz klase *Professional*
* *art\_style* (*string*): umjetnički stil kojim se umjetnik bavi
* *awards* (*int*): broj nagrada koje je umjetnik dobio

Metode klase *Artist*:

* *\_\_init\_\_*(*self, first\_name, last\_name, age, expertise\_level, salary, nationality, art\_style, awards*): konstruktor klase *Artist*
* *personal\_info*(*self*) -> *str*: **polimorfizam**; vraća osnovne informacije o umjetniku s detaljem vezanim za umjetnike (atribut *art\_style*)
* *achievements*(*self*) -> *str*: **polimorfizam**; vraća postignuća umjetnika, specifična za umjetnike
* *create\_masterpiece*(*self*) -> *str*: vraća poruku o stvaranju remek-djela
* *practice*(*self*) -> str: vraća poruku o tome kako umjetnik provodi vrijeme vježbajući
* *perform\_action*(*self*) -> *str*: **polimorfizam**; vraća radnju koju umjetnik izvodi, specifičnu za umjetnike

Klasa Scientist

Klasa *Scientist* također nasljeđuje klasu *Professional* i predstavlja osobu koja se bavi znanošću.

Atributi klase *Scientist*:

* svi atributi klase *Professional*
* *field* (*string*): područje istraživanja znanstvenika
* *publications* (*int*): broj radova koje je znanstvenik objavio

Metode klase *Scientist*:

* *\_\_init\_\_*(*self, first\_name, last\_name, age, expertise\_level, salary, nationality, field, publications*): konstruktor klase *Scientist*
* *personal\_info*(*self*) -> *str*: **polimorfizam**; vraća osnovne informacije o znanstveniku s detaljem vezanim za znanstvenike (atribut *field*)
* *achievements*(*self*) -> *str*: **polimorfizam**; vraća postignuća znanstvenika, specifična za znanstvenike
* *conduct\_experiment*(*self*) -> *str*: vraća poruku o provođenju eksperimenata
* *analyze\_data*(se*lf*) -> *str*: vraća poruku o analizi podataka
* *perform\_action*(*self*) -> *str*: **polimorfizam**; vraća radnju koju znanstvenik izvodi, specifičnu za znanstvenike

Klasa *Entrepreneur*

Klasa *Entrepreneur* također nasljeđuje klasu *Professional* i predstavlja osobu koja se bavi poduzetništvom.

Atributi klase *Entrepreneur*:

* svi atributi klase *Professional*
* *industry* (*string*): industrija u kojoj poduzetnik djeluje
* *ventures* (*int*): broj poduzetničkih pothvata koje je poduzetnik poduzeo

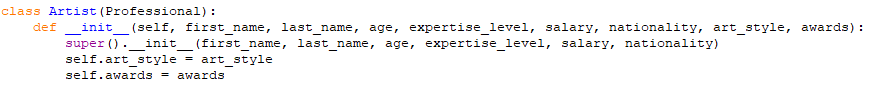
Metode klase *Entrepreneur*:

* *\_\_init\_\_*(*self, first\_name, last\_name, age, expertise\_level, salary, nationality, industry, ventures*): konstruktor klase *Entrepreneur*
* *personal\_info*(*self*) -> *str*: **polimorfizam**; vraća osnovne informacije o poduzetniku s detaljem vezanim za poduzetnike (atribut *industry*)
* *achievements*(*self*) -> *str*: **polimorfizam**; vraća postignuća poduzetnika, specifična za poduzetnike
* *launch\_startup*(*self*) -> *str*: vraća poruku o pokretanju startupa
* *negotiate\_deals*(*self*) -> *str*: vraća poruku o pregovaranju oko poslova
* *perform\_action*(*self*) -> *str*: **polimorfizam**; vraća radnju koju poduzetnik izvodi, specifičnu za poduzetnike

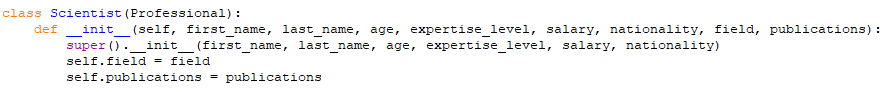
Principi objektno orijentiranog programiranja (nasljeđivanje i polimorfizam):

1. Nasljeđivanje

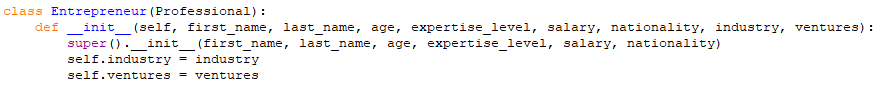
Klasa *Artist* nasljeđuje klasu *Professional*, što omogućuje umjetnicima pristup zajedničkim atributima i metodama „profesionalnim“ osobama općenito, ali i dodavanje specifičnih atributa i metoda za umjetnike.



Klasa Scientist nasljeđuje klasu Professional, što omogućuje znanstvenicima pristup zajedničkim atributima i metodama "profesionalnim" osobama općenito, ali i dodavanje specifičnih atributa i metoda za znanstvenike.



Klasa Entrepreneur nasljeđuje klasu Professional, što omogućuje poduzetnicima pristup zajedničkim atributima i metodama "profesionalnim" osobama općenito, ali i dodavanje specifičnih atributa i metoda za poduzetnike.



1. Polimorfizam

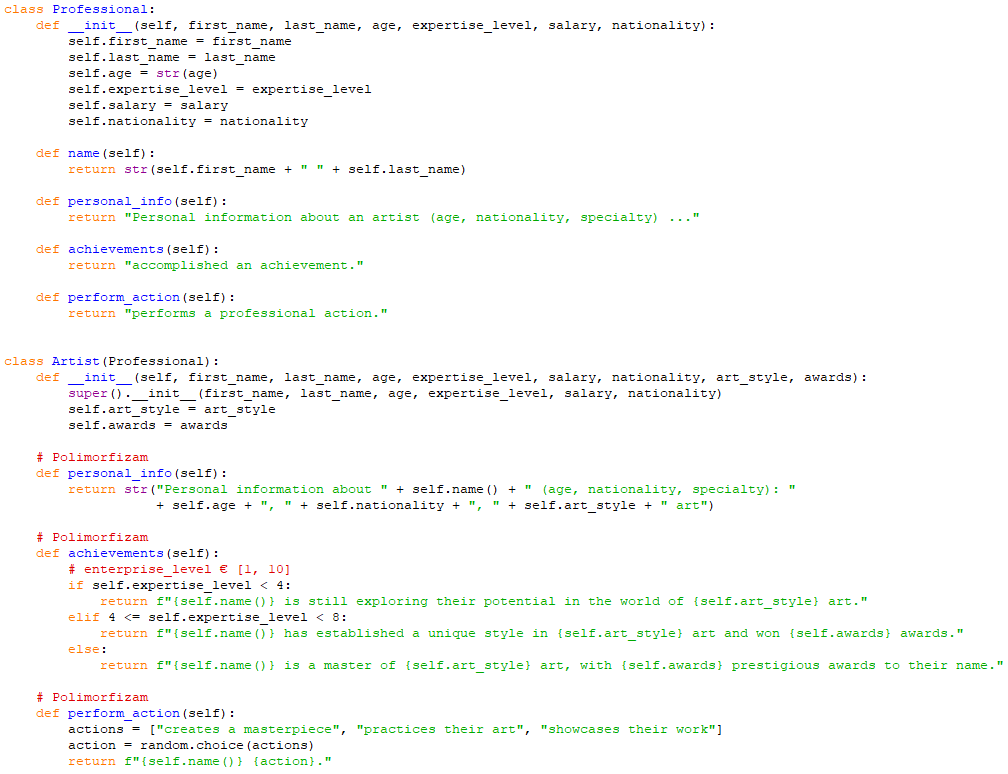
Metode *personal\_info*(), *achievements*() i *perform\_action*() polimorfne su u sve tri klase koje nasljeđuju roditeljsku klasu *Professional*: *Artist*, *Scientist* i *Entrepreneur*, pružajući različito ponašanje ovisno o tipu objekta koji ih poziva.

U metodi *personal\_info*() razlika je u ispisu jednog atributa (*art\_style*/*field*/*industry*) ovisno o tipu objekta koji je poziva.

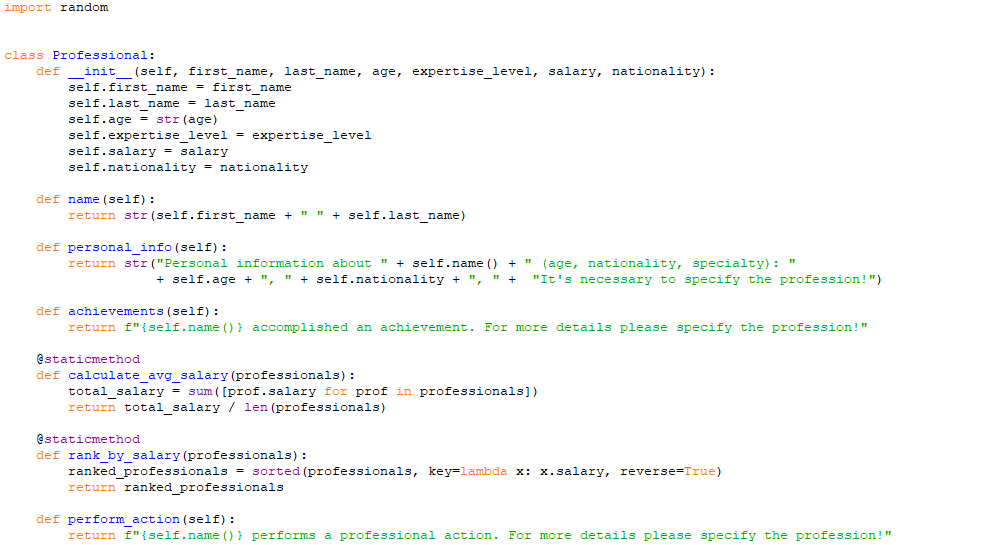
U metodi *achievements*() razlika je u razini stručnosti (*enterprise\_level*) koja se koristi za provjeru u uvjetima metode, a to također ovisi o tipa objekta koji ovu metodu poziva.

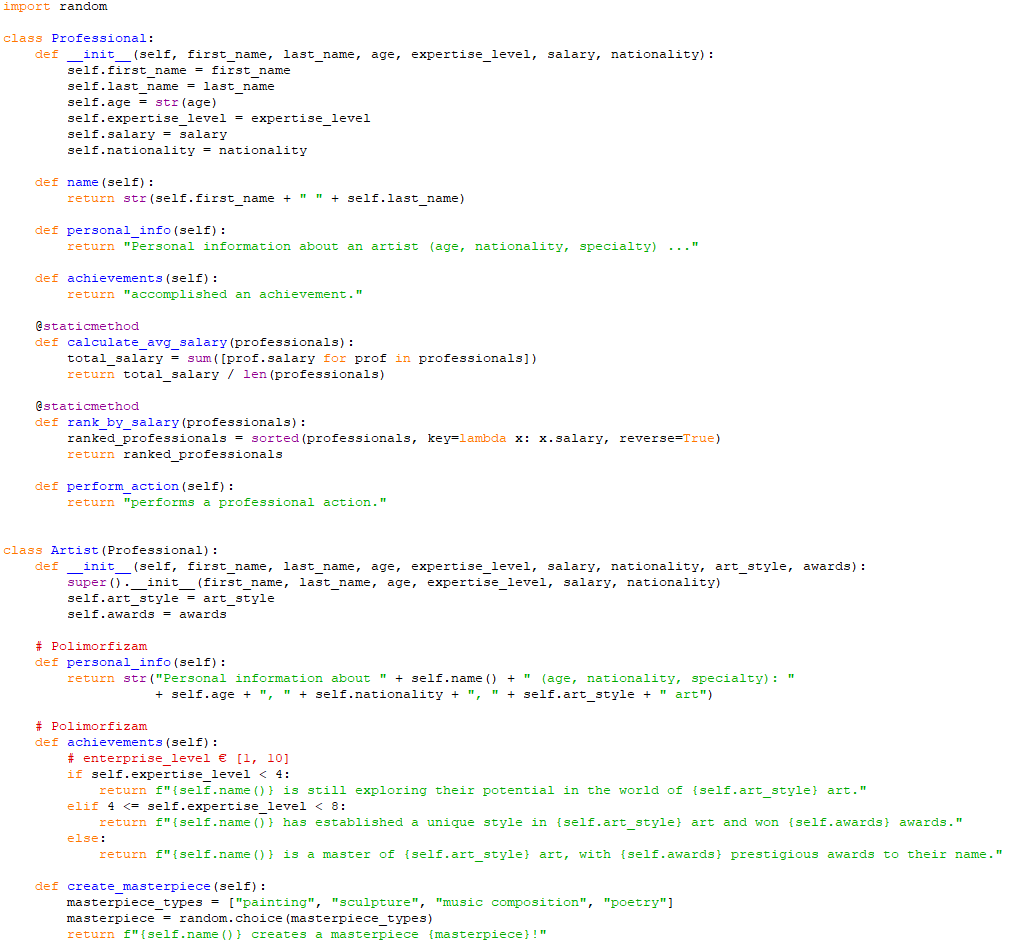
U metodi *perform\_action*() o vrsti radnji koju određena osoba izvodi, a to kao i za gornje dvije polimorfne metode ovisi o vrsti objekta koji ovu metodu poziva.

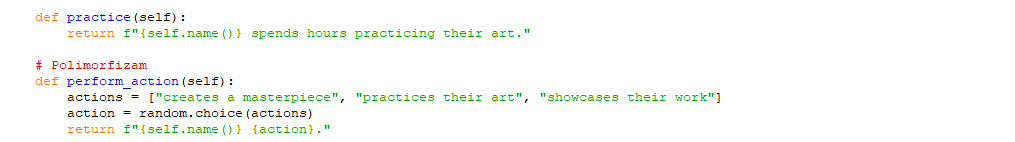
Primjer polimorfizma između roditeljske klase Professional i klase Artists koja ju nasljeđuje (cijeli se kod može vidjeti na kraju ovoga dokumenta, kao i svi polimorfizmi):



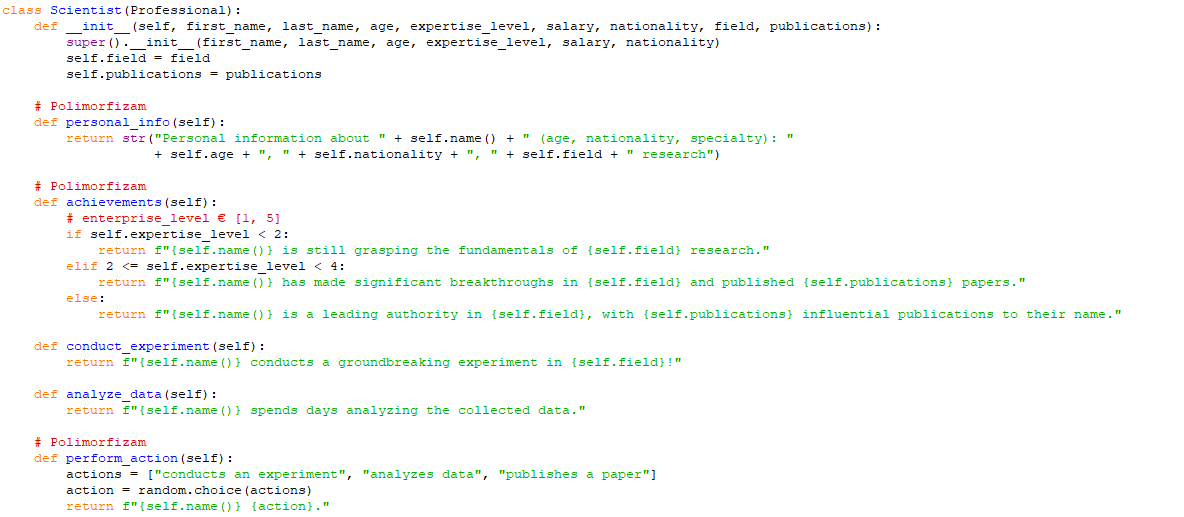
Prikaz koda skripte *professional.py*:

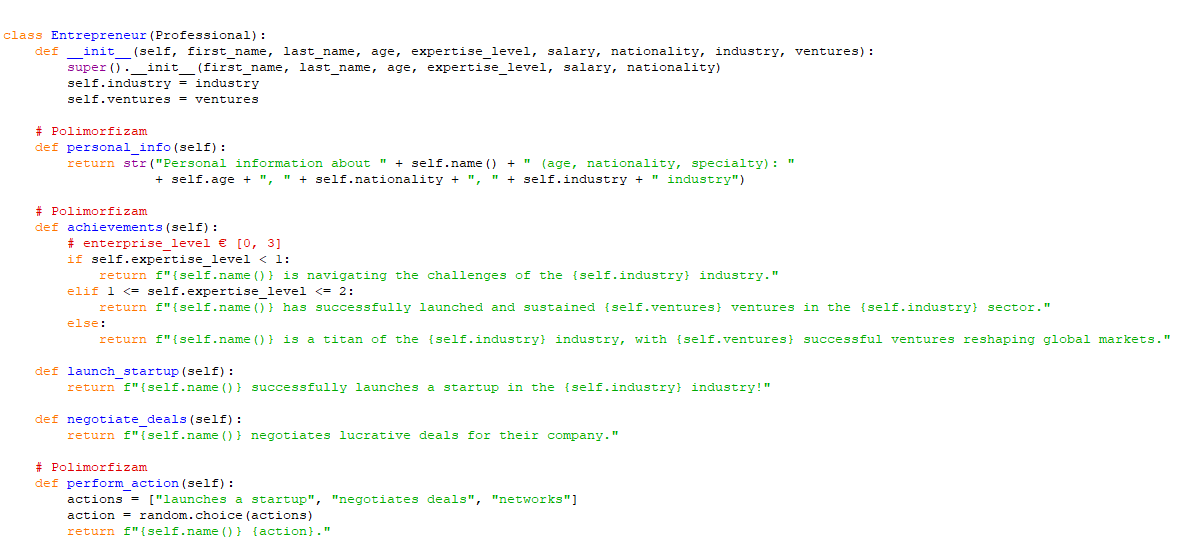






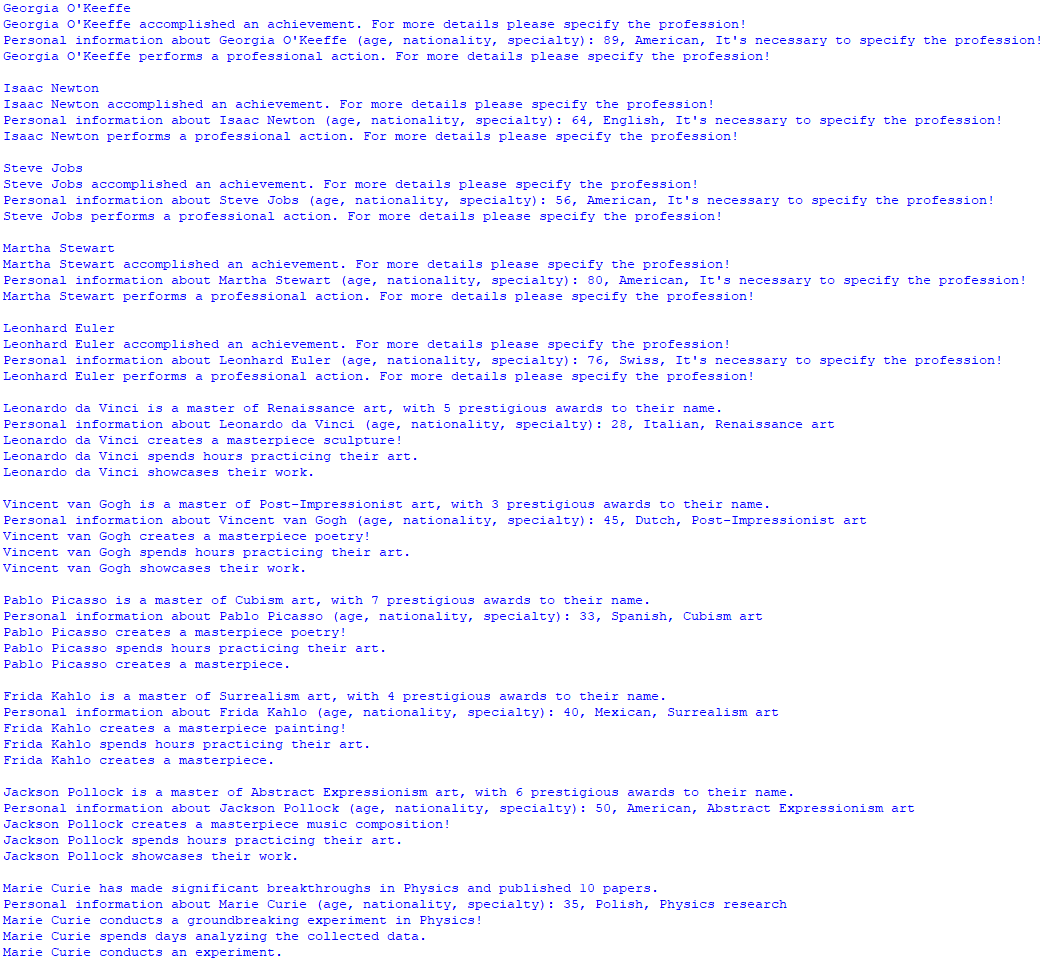
Prikaz koda skripte *professional.py* (nastavak):

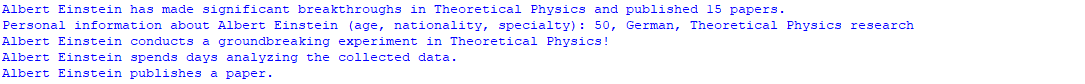




Ispis nakon pokretanja skripte *main.py* (skripta nije prikazana u ovom dokumentu, može se pronaći u predanoj .*zip* mapi):

* u skripti *main.py* može se vidjeti inicijalizacija traženih pet objekata za svaku klasu i pozivi svih metoda u klasama





Ispis nakon pokretanja skripte *main.py* (nastavak):

