Pattern Recognition
and Applications Lab

Le basi della Programmazione Orientata agli Oggetti (Esercizi)

Docente: Ambra Demontis

Anno Accademico: 2024 - 2025

University of Cagliari, Italy

Department of Electrical and Electronic Engineering

Corso di Laurea in Ingegneria Elettronica, Informatica e delle Telecomunicazioni

Voto Finale Esami (versione 1)

Progettiamo un programma che permetta ad uno studente di memorizzare, per ogni esame, i voti presi in due esami parziali e che metta a disposizione un metodo che permetta di mostrare, per ognuno degli esami, il voto finale, che è dato dalla media aritmetica dei voti conseguiti nei due parziali.

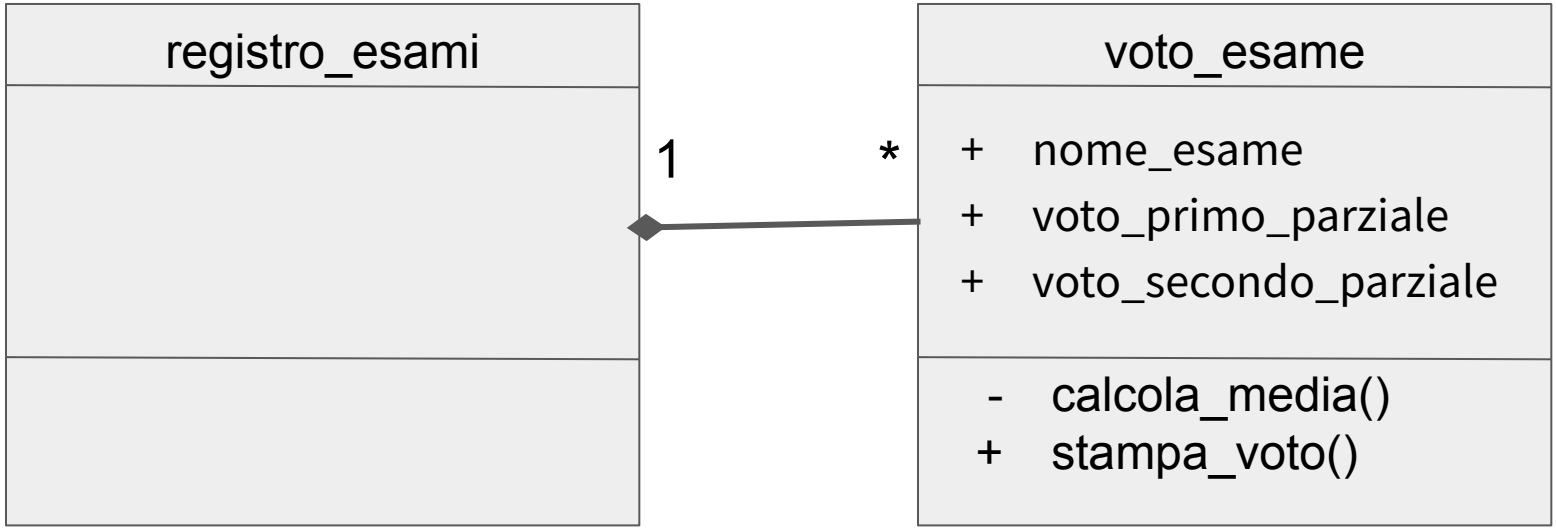
Di quali Oggetti Abbiamo Bisogno?

Di quali Oggetti Abbiamo Bisogno?

1) Un oggetto che contenga i voti di un esame
2) Un oggetto che ci permetta di memorizzare i voti di tanti esami

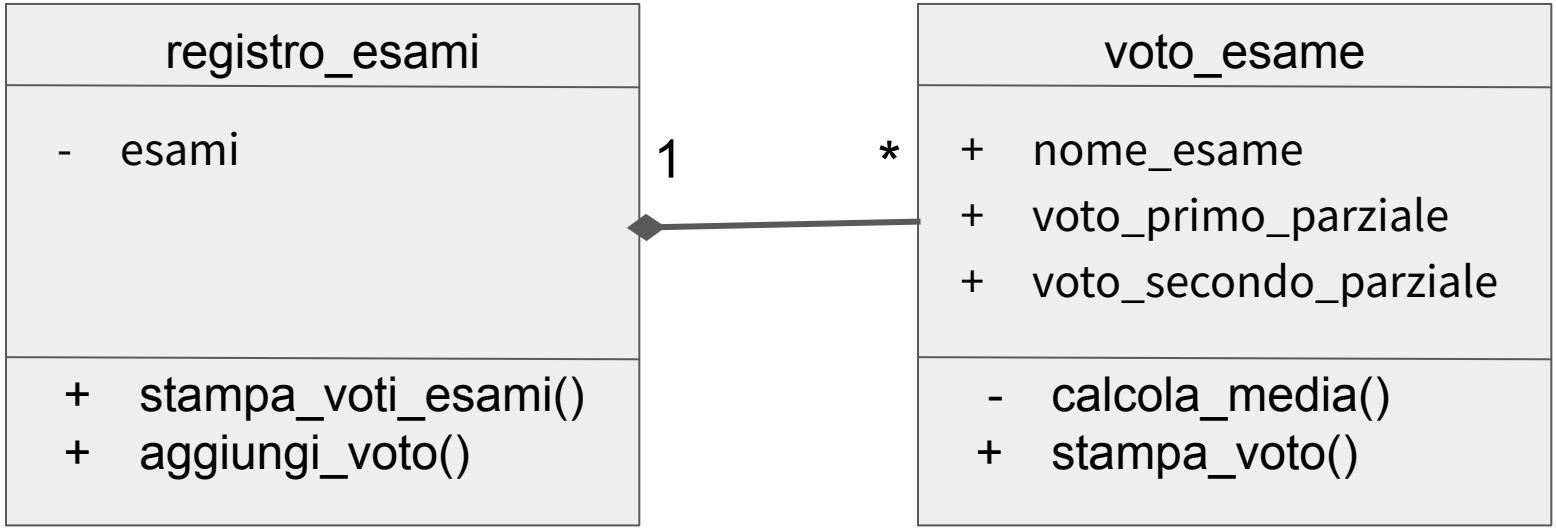
La classe Voti_esame

La Relazione tra le classi



La classe Lista_voti

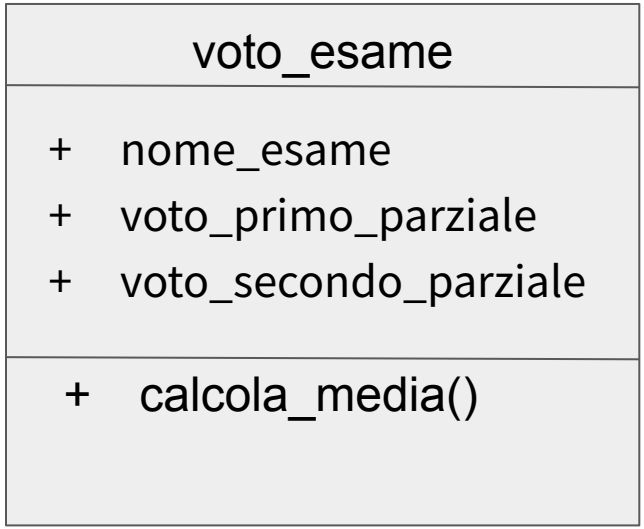
La Relazione tra le classi



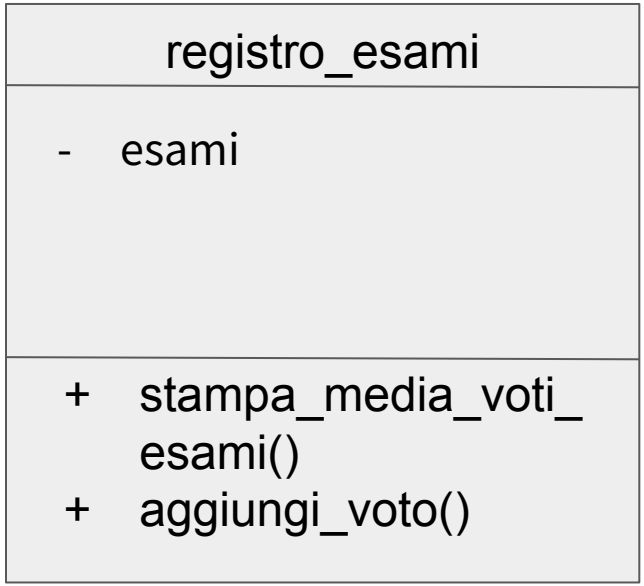
Voto Finale Esami (versione 2)

Progettiamo un programma che permetta ad uno studente di memorizzare, per ogni esame, i voti presi in due esami parziali. *Il programma deve mettere a disposizione inoltre un metodo che permetta di mostrare, la media aritmetica ottenuta in tutti gli esami*. Il voto finale di un singolo esame supponiamo sia dato dalla media aritmetica dei voti conseguiti nei due parziali.

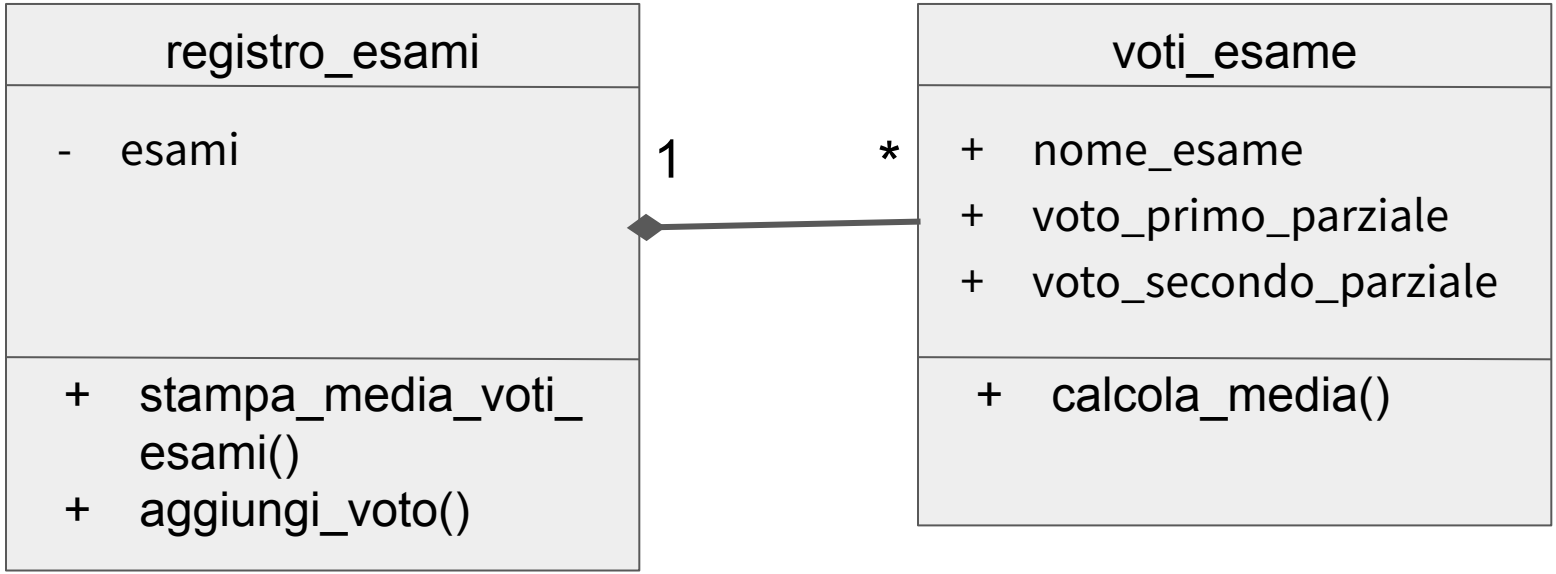
La classe Voti_esame



La classe Lista_voti



La Relazione tra le classi



Il Programma per l'Anagrafica Clienti di un Negozio

Supponiamo di essere un programmatore al quale un negoziante ha chiesto di creare un programma per la gestione delle informazioni anagrafiche dei suoi clienti (che possono essere aziende o persone).

Il programma deve permettere al negoziante di:

1) Memorizzare i dati dei suoi clienti.

    Per le persone: idx, nome, cognome, anno di nascita, numero di telefono.

    Per le aziende: idx, partita iva, numero di telefono.

    Dove idx è un identificativo univoco assegnato dal negozio ai suoi clienti.

2) Poter cercare un cliente utilizzando l’idx.

Quali Classi ci Servono?

Quali Classi ci Servono?

1) Dobbiamo memorizzare i dati dei clienti, che sono differenti a seconda del fatto che il cliente sia un’azienda o una persona.

1) Entrambe le classi azienda e persona hanno un attributo idx.

1) Dobbiamo poter memorizzare tante aziende e tante persone quindi ci serve un oggetto che le contenga.

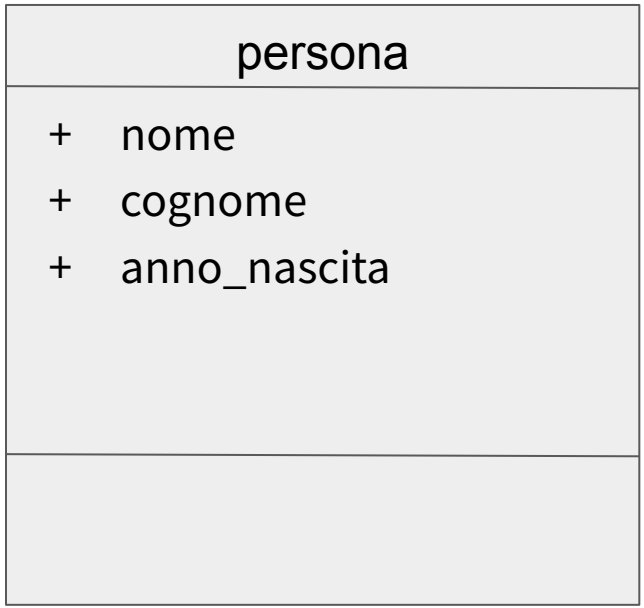
1) Dobbiamo poter cercare i clienti tramite il loro attributo idx

Quali Classi ci Servono?

1) Dobbiamo memorizzare i dati dei clienti, che sono differenti a seconda del fatto che il cliente sia un'azienda o una persona.
   Avremo due classi: *azienda* e *persona*
1) Entrambe le classi azienda e persona hanno un attributo idx.
   Possiamo creare una classe *cliente* con un attributo idx e numero telefono
1) Dobbiamo poter memorizzare tante aziende e tante persone quindi ci serve un oggetto che le contenga.
   Avremo una classe *lista_clienti*
1) Dobbiamo poter cercare i clienti tramite il loro attributo idx
   Può occuparsene sempre la classe *lista_clienti*

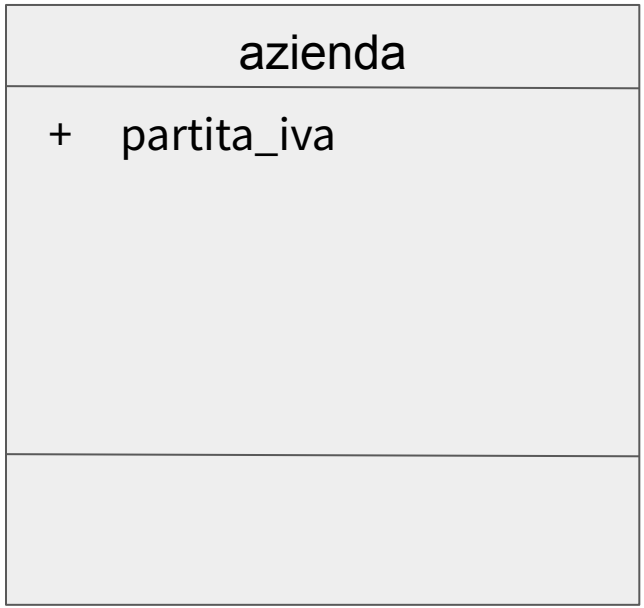
La Classe Persona

Deve contenere i dati di una persona.



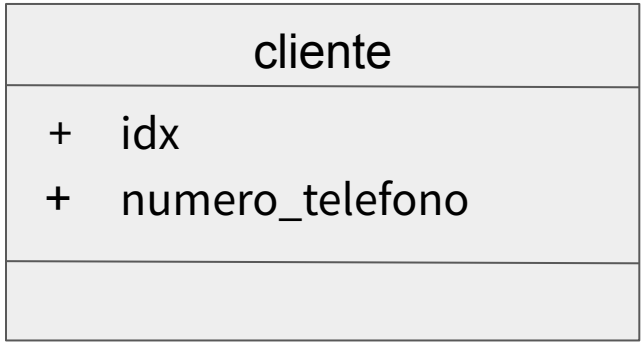
La Classe Azienda

Deve contenere i dati di un’azienda.

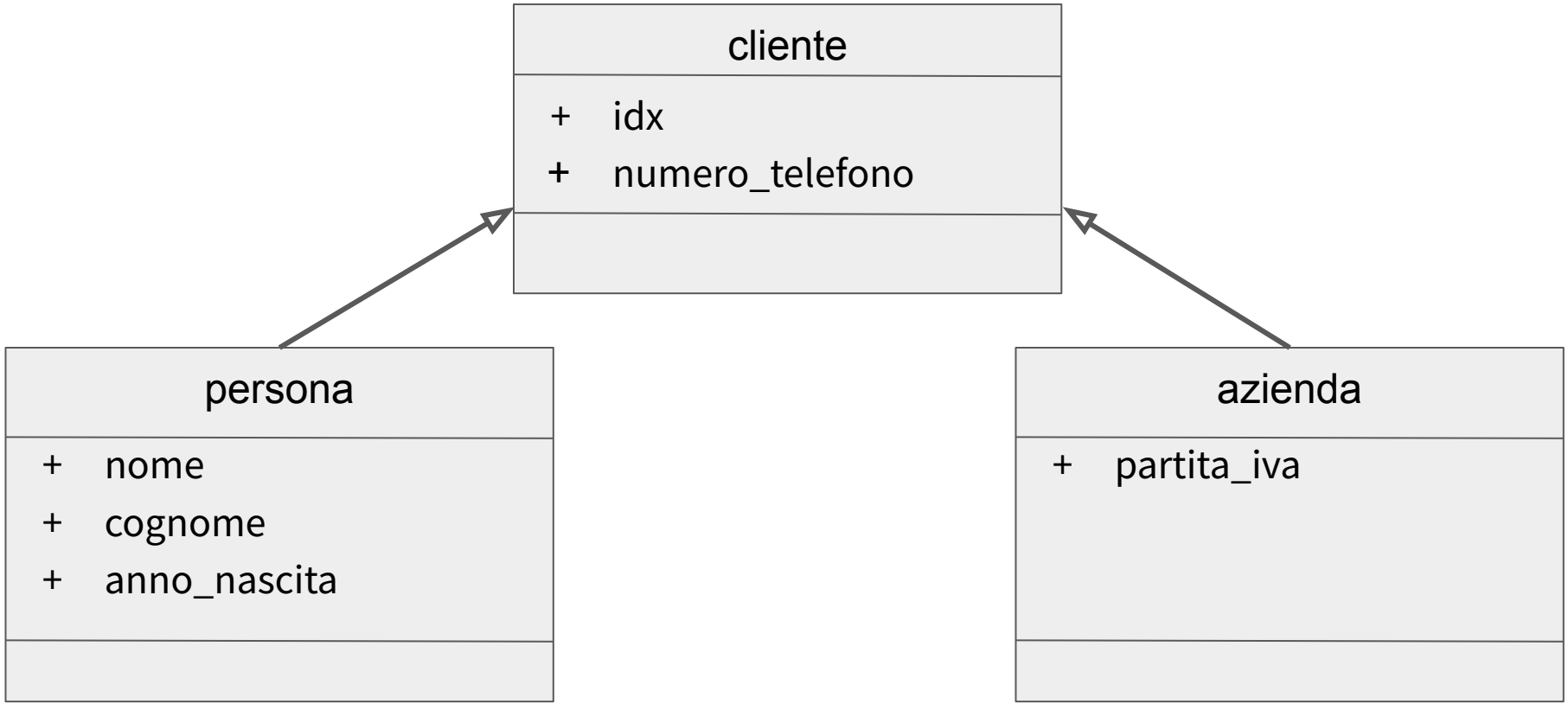


La Classe Cliente

E’ la classe padre delle classi *azienda* e *persona*.
Contiene attributi e metodi in comune alle due classi, nel nostro caso, l’attributo *idx* e *numero_telefono*.



La Relazione di Ereditarietà

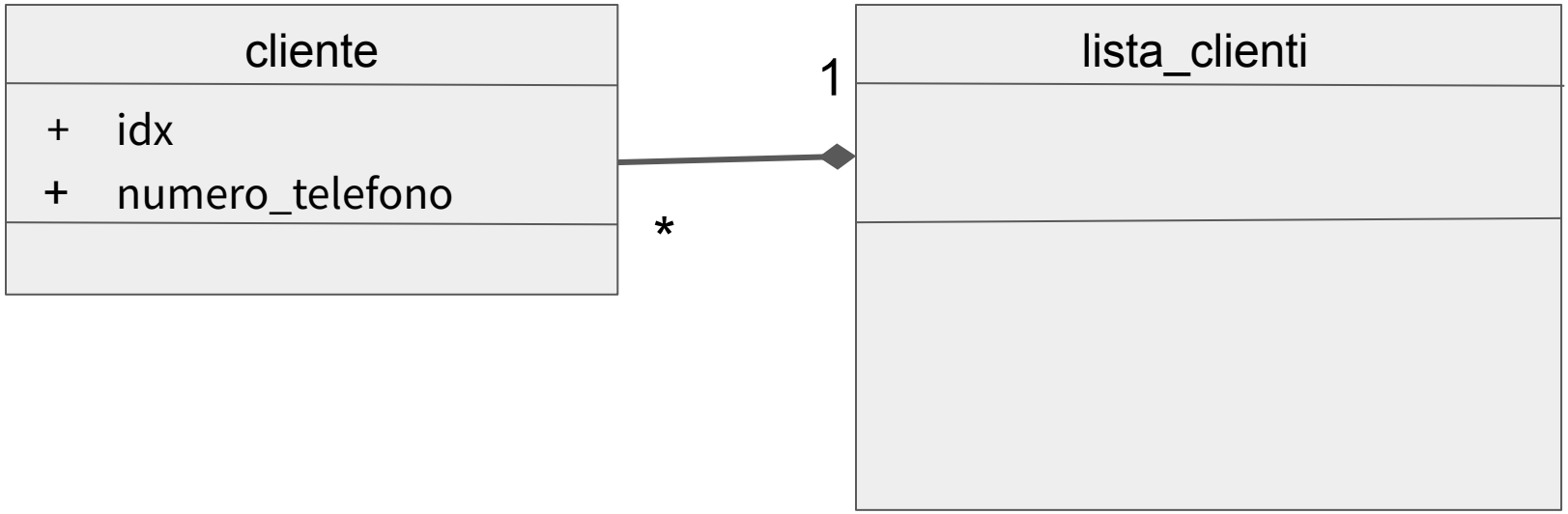


La classe Lista_clienti

Deve memorizzare clienti, che possono essere di tipo *persona* o *azienda* e permettere di cercare clienti.

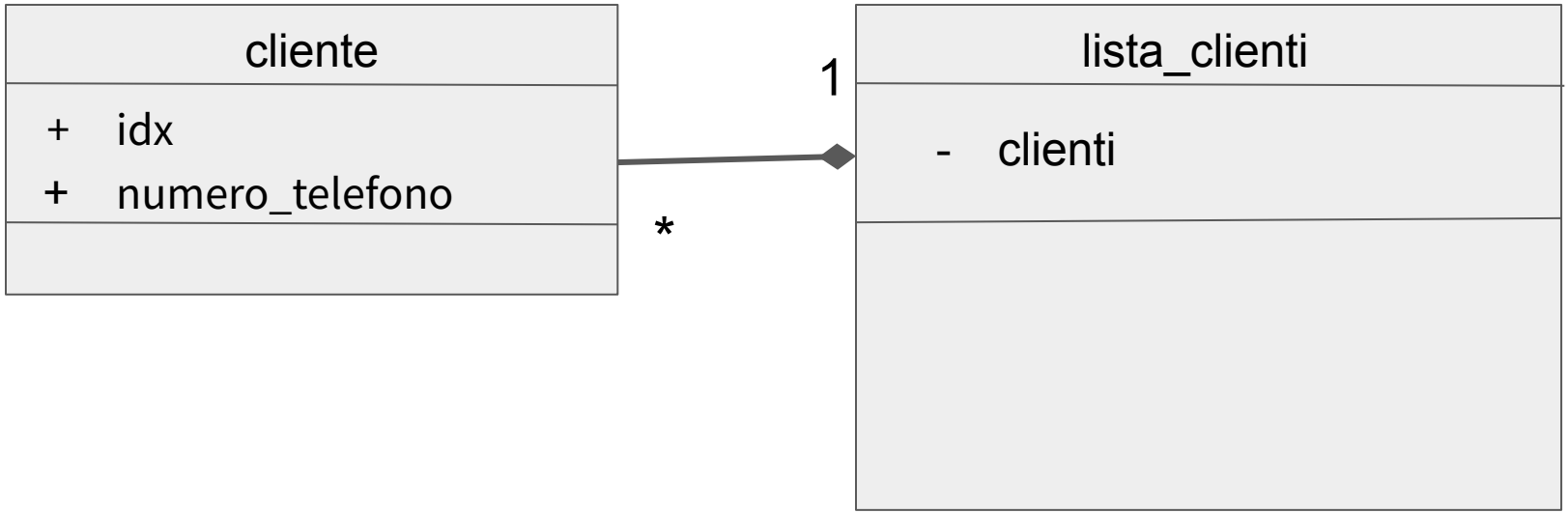
Relazioni dell’Oggetto Lista_clienti

E’ composta da oggetti di classe cliente.

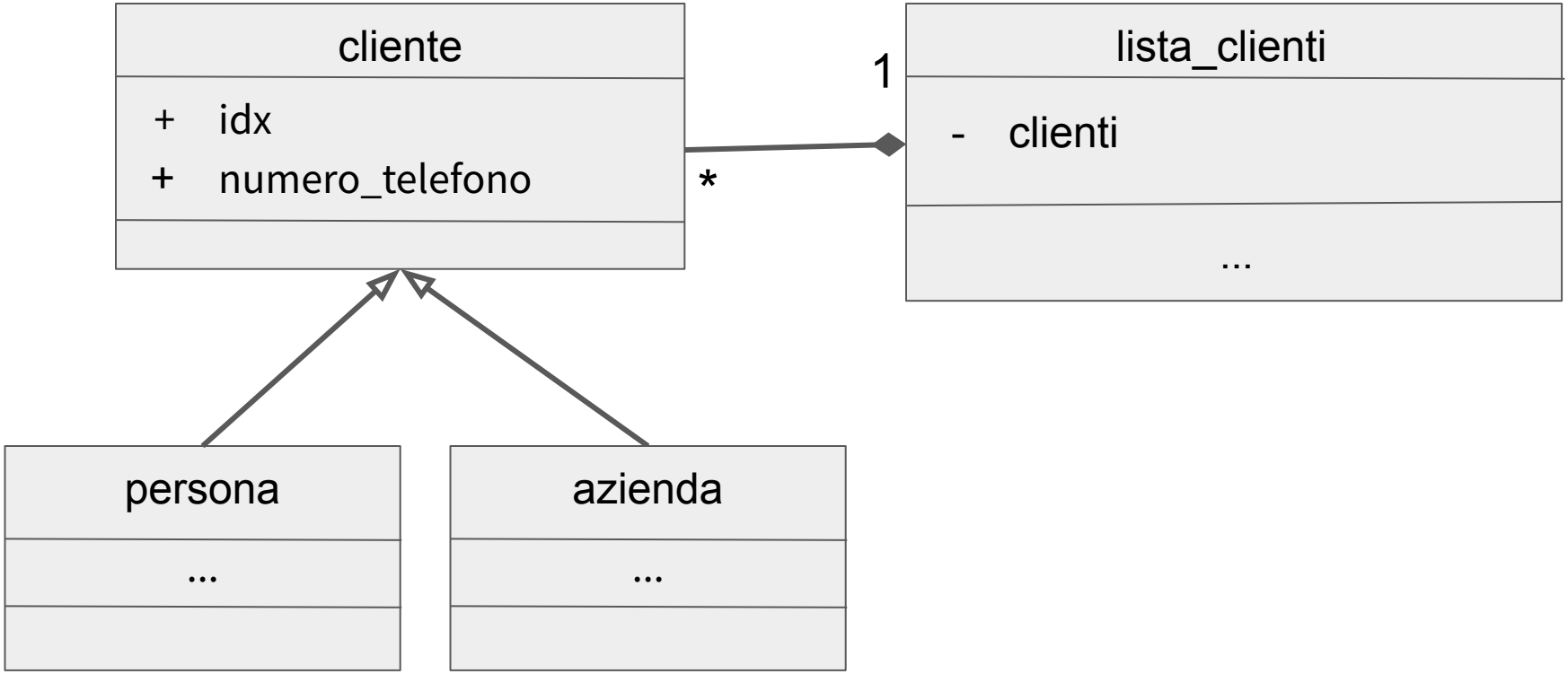


Relazioni dell’Oggetto Lista_clienti

E’ composta da oggetti di classe cliente.



Riepiloghiamo Tutte le Relazioni

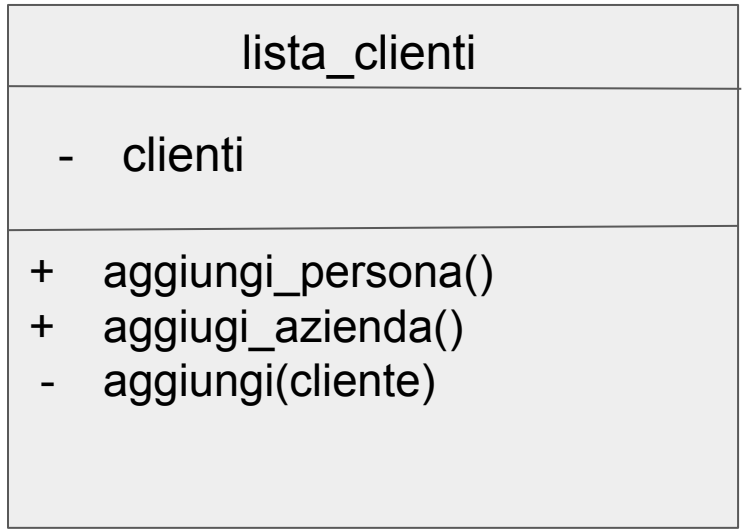


La classe Lista_clienti

Deve memorizzare clienti, che possono essere di tipo *persona* o *azienda* e permettere di cercare clienti.

Funzioni pubbliche che creano un oggetto di tipo persona o azienda e usano la funzione privata aggiungi per memorizzarlo

Funzione privata che riceve e memorizza un oggetto di tipo cliente



La classe Lista_clienti

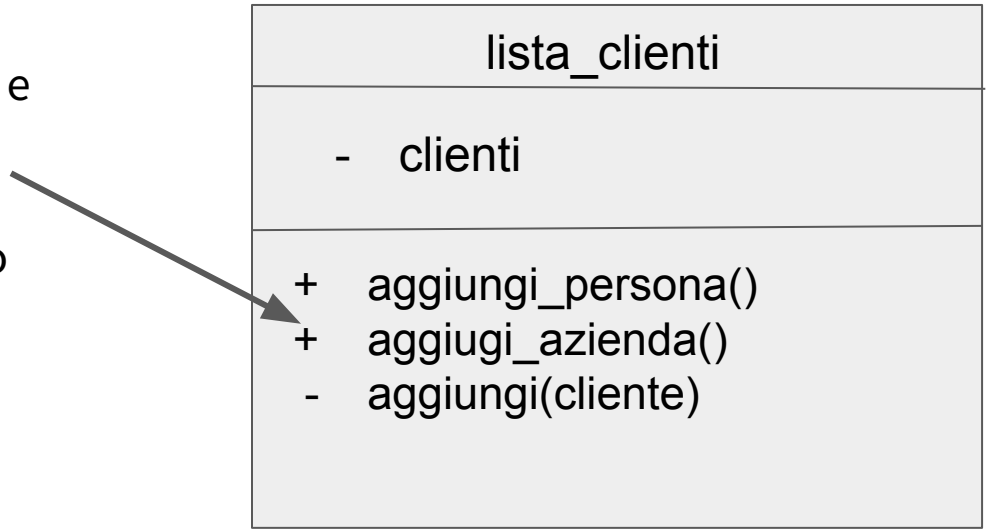
Deve memorizzare clienti, che possono essere di tipo *persona* o *azienda* e permettere di cercare clienti.

Nb: le funzioni *aggiugi_persona* e *aggiungi_azienda* potrebbero occuparsi di acquisire anche in input i valori degli attributi oppure potremmo avere in quelle due classi una funzione dicata es. *acquisisci_attributi*

La classe Lista_clienti

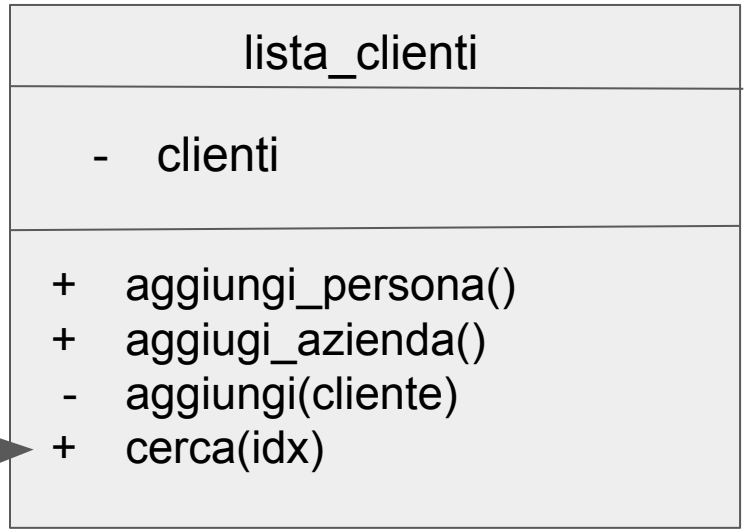
Deve memorizzare clienti, che possono essere di tipo *persona* o *azienda* e permettere di cercare clienti.

In ogni caso, il comportamento delle funzioni *aggiungi_persona* e *aggiungi_azienda* non sarebbe completamente uguale perchè devono creare un oggetto di tipo differente.



La classe Lista_clienti

Deve memorizzare clienti, che possono essere di tipo *persona* o *azienda* e permettere di cercare clienti.



Funzione pubblica che ci permette di cercare un cliente utilizzando il suo idx