

Programação Orientada à Objetos

Reutilização de Classes

- Associação de classes (agregação/composição)
- Multiplicidade ?
 - Define a quantidade de instâncias de um objeto que uma classe pode ter em relação a outra classe

Multiplicidades:

Nome	Simbologia
Apenas Um	1..1 (ou 1) (ou em branco)
Zero ou Muitos	0..* (ou *)
Um ou Muitos	1..*
Zero ou Um	0..1
Intervalo Específico	$l_i..l_s$

Programação Orientada à Objetos:

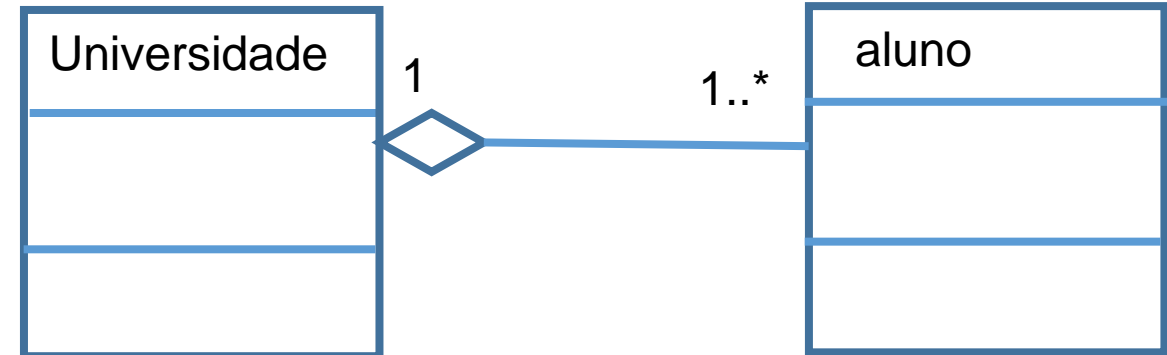
Reutilização de Classes

Multiplicidade Obrigatória Um para N



Significado: A classe da extremidade esquerda deve ter no mínimo UMA e no máximo N ocorrências do objeto definido pela classe da extremidade direita

Ex:



Programação Orientada à Objetos: Reutilização de Classes

Implementação de associação de classes com
multiplicidades em python...

Programação Orientada à Objetos: Reutilização de Classes

```
class Universidade:
    def __init__(self, sigla, nome, tipo):
        self.__sigla=sigla
        self.__nome=nome
        self.__tipo=tipo # federal, estadual ou particular
        self.__alunos=[]

    @property
    def sigla(self):...

    @property
    def nome(self):...

    def cadastrarAluno(self, aluno):...

    def excluirAluno(self, aluno):...

    def __str__(self):...
```

Programação Orientada à Objetos:

Reutilização de Classes

```
class Aluno:
    def __init__(self, cpf, nome, origem):
        self.__cpf=cpf
        self.__nome=nome
        self.__origem=origem #sisu, prouni, portador diploma

    def __str__(self):
        return(self.__nome+"; "+self.__origem)

    @property
    def nome(self):
        return self.__nome
```