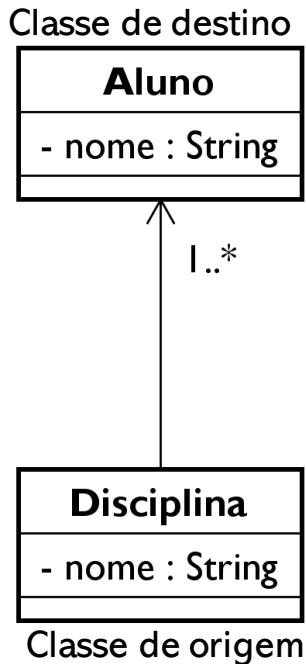


# **Associações com multiplicidade acima de 1**

# Passos para tradução da associação



## 1. Deve-se criar uma variável na classe de origem

O identificador da variável será igual ao papel. Na ausência, o nome do identificador deve ser derivado do nome da classe de destino. Geralmente, é igual ao nome da classe, mas no plural.

## 2. O tipo de dado desta variável deve ser `ArrayList`

Utilizar o operador diamante informando o nome da classe de destino

## 3. Criar a instância da classe `ArrayList`

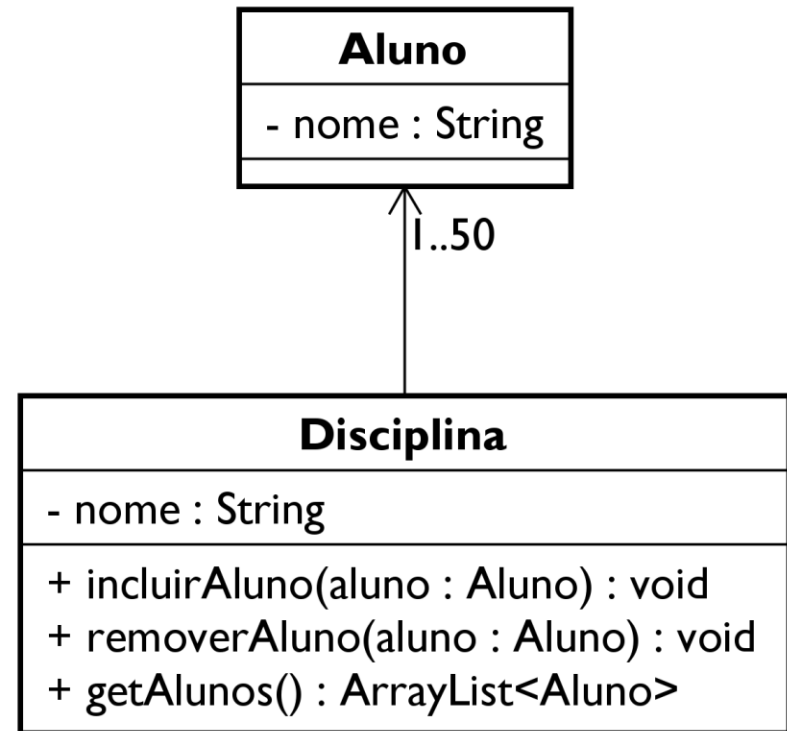
Pode ser feito na declaração da variável ou no construtor.

```
3  import java.util.ArrayList;
4
5  public class Disciplina {
6
7      private String nome;
8      private ArrayList<Aluno> alunos = new ArrayList<>();
9  }
```

Exemplo de tradução

# Passos para tradução da associação

4. A classe de origem deve implementar métodos para **incluir**, **remover** e **obter** os objetos contidos na associação {



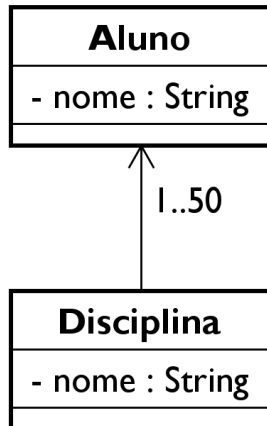
# Passos para tradução da associação

## Método para incluir alunos na disciplina

Deve receber como parâmetro um objeto a ser adicionado

```
public void incluirAluno(Aluno aluno) {  
    alunos.add(aluno);  
}
```

Se a associação for limitada, deve-se impedir incluir mais objetos do que o permitido



```
public void incluirAluno(Aluno aluno) {  
    if (alunos.size() == 50) {  
        throw new RuntimeException("Não é possível incluir "  
            + "mais alunos na disciplina");  
    }  
    alunos.add(aluno);  
}
```

# Passos para tradução da associação

## Método para remover alunos na disciplina

Deve receber como  
parâmetro um objeto a  
ser removido



```
public void removerAluno(Aluno aluno) {  
    alunos.remove(aluno);  
}
```

# Passos para tradução da associação

*Getter* para a variável que mantém a associação

```
public ArrayList<Aluno> getAlunos() {  
    return alunos;  
}
```

**Não** existe setter para a variável que mantém a associação