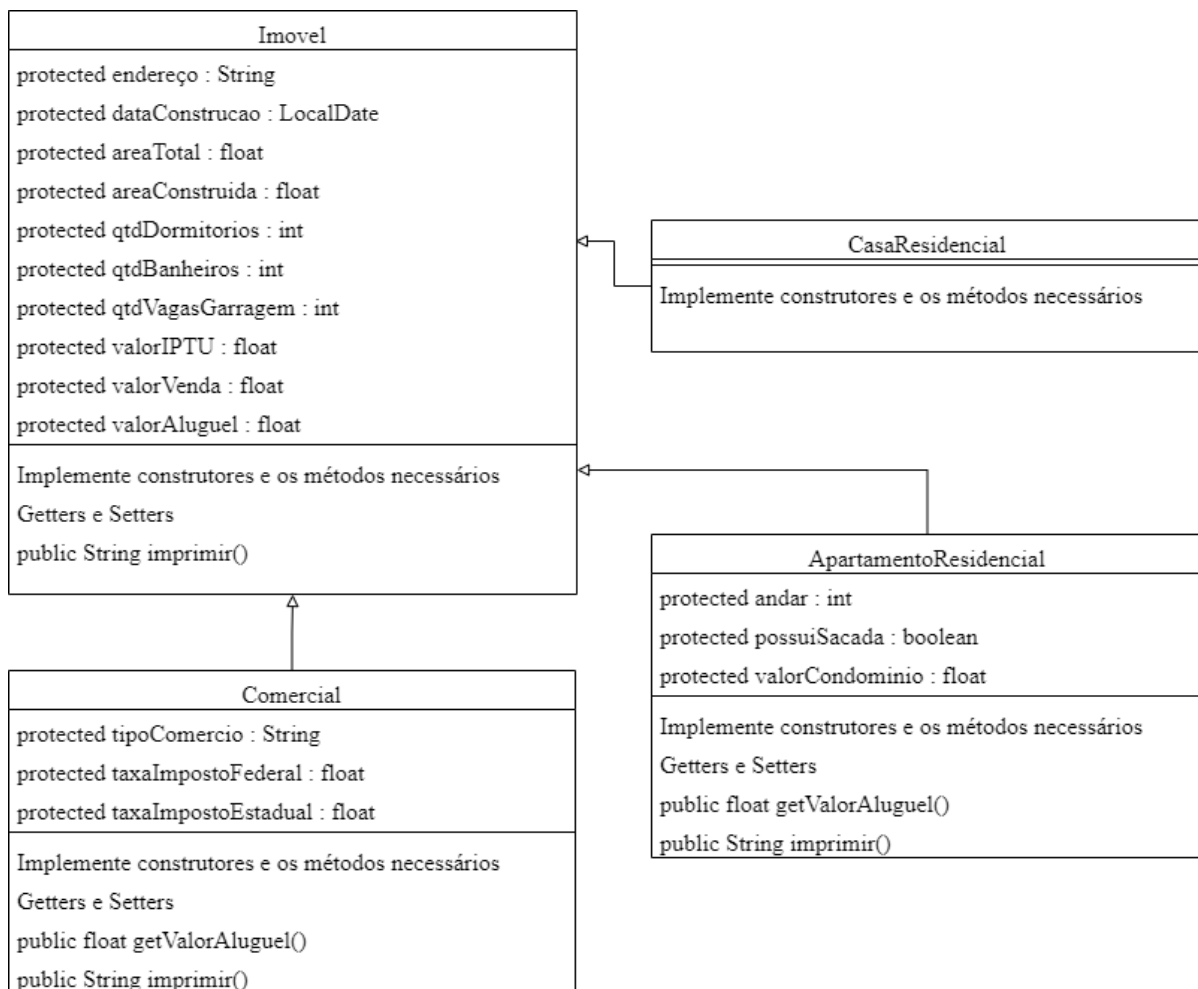


**Faculdade de Tecnologia de Presidente Prudente**  
**Análise e Desenvolvimento de Sistemas - AMS**  
**Inteligência Artificial e Aprendizado de Máquina**

**Lista de Exercícios**  
**Programação Orientada à Objetos - Python**

**OBS: Em todos os exercícios faça algumas instâncias dos objetos modelados pelas classes criadas**

- 1) Na disciplina de Modelagem de Padrões de Projetos você e seu grupo estão desenvolvendo um Sistema WEB que faz uso de Orientação à Objetos. Implemente em Python todas as Classes da camada de negócio existentes em seu projeto.
- 2) Implemente uma Classe Imóvel que possui três Subclasses: ApartamentoResidencial, CasaResidencial e Comercial. Para isso, considere as informações a seguir e a representação UML para classes.



- 3) Implemente uma Classe Professor e uma Classe Departamento. Para isso, considere as informações a seguir e a representação UML para classes.

### Professor

Atributos: matricula, nome, titulação (ex: mestre, doutor, ...), telefone, E-mail, salário.

Métodos:

- ✓ Construtores
- ✓ *Getters()* e *Setters()*
- ✓ *Imprime()* : Exibe todos os dados de um professor, ou seja os valores de seus atributos.

### Departamento:

Atributos: Nome (Ex: Departamento de Computação), Telefone, E-mail, Lista de Professores do departamento (Objetos da classe Professor), sugestão: Utilize por exemplo um Lista, Vetor ou alguma estrutura de dados.

Métodos:

- ✓ Construtores
- ✓ *Getters()* e *Setters()*
- ✓ *adicionarProfessor()* : Recebe como parâmetro um Objeto professor e o adiciona na lista de professores.
- ✓ *excluirProfessor()* : Recebe como parâmetro o número da matrícula de um professor, com base em sua matrícula, excluir da lista de professores o professor com a matrícula correspondente.
- ✓ *buscaProfessor()* : Recebe como parâmetro o número da matrícula de um professor, com base em sua matrícula, busca na lista de professores o professor com a matrícula correspondente. Retorna *true* caso encontre e *false* caso não encontre.
- ✓ *Imprime()* : Exibe todos os dados de um departamento, ou seja, valores de seus atributos. Além disso, exibe também todos os Professores presentes na lista de professores. (Obs: é exibido os dados de todas os Professores presentes na lista).

Professor
<pre>private matricula : int private nome : String private titulacao : String private telefone : String private email : String private salario : double</pre>
<pre>Professor() Professor(int matricula, String nome, String titulacao, String telefone, String email, double salario) public void imprime()</pre>

Departamento
<pre>private nome : String private telefone : String private email : String private ArrayList&lt;Professor&gt; Professores</pre>
<pre>Departamento() Departamento(String nome, String telefone, String email) public void adicionarProfessor(Professor professor) public boolean buscarProfessor(int matricula) public void excluirProfessor(int matricula) public void imprime()</pre>

**ENTREGAR PELA PLATAFORMA TEAMS ATÉ O DIA 21/04**