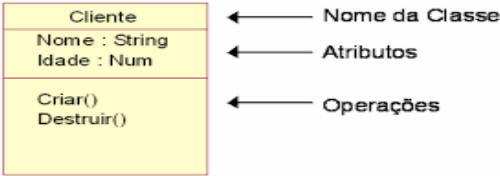
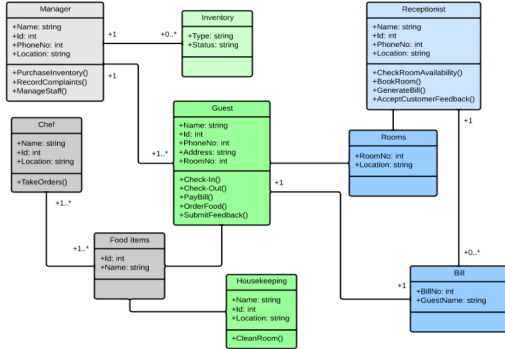



VISIBILIDADE DOS OBJETOS		VISIBILIDADE DOS OBJETOS
O que é a UML ?		UML: é a Unified Modelling Language (Linguagem de Modelagem Unificada), ela é uma espécie de padrão de modelagem para o desenvolvimento de softwares. A UML traz uma série de conceitos, ferramentas e diagramas que podem ser implementados para ajudar na visualização dos elementos de um sistema. Para a área de POO especificamente, o diagrama mais importante é o Diagrama de Classes.
O que é o Diagrama de Classes ?		Diagrama de Classes: é um diagrama específico para a representação das classes durante a modelagem de um projeto de software. Essa diagrama torna possível a boa visualização das classes, seus atributos e métodos. Ele segue o seguinte padrão: - Classes: Representada pelo Retângulo, seu nome fica posicionado num container no topo do retângulo, por convenção as classes devem ser CamelCase; - Atributos: Ficam posicionados no container abaixo do nome da Classe, por convenção os atributos são escritos em letra minúscula; - Métodos: Ficam posicionados no container mais baixo do retângulo, por convenção são escritos em letra minúscula e ao final sempre teremos os parênteses representando a entrada de argumentos dos métodos;
Ilustre a representação de uma Classe dentro de um Diagrama de Classes:		
Ilustre um Diagrama de Classes completo :		
<i>- reservada para a questão acima -</i>		
O que é o conceito de visibilidade dentro da POO?		O conceito de Visibilidade em POO consiste no quanto os elementos de uma Classe - atributos e métodos - são acessíveis do lado de fora da Classe. Podemos dividir os níveis de acessibilidade em 3, que são: - Públicos: Elementos que PODEM SER acessados de fora da Classe, eles são representados em UML pelo símbolo de "+"; - Privados: Elementos que NÃO PODEM SER acessados de fora da Classe, eles são representados em UML pelo símbolo de "-"; - Protegidos: Elementos que SÓ PODEM SER ACESSADOS PELAS SUBCLASSES DE UMA CLASSE, em UML eles são representados pelo símbolo de "#";
<i>- reservada para a questão acima -</i>		
Ilustre como podemos usar as visibilidades numa Classe dentro de um Diagrama de Classes:		
<i>- reservada para a questão acima -</i>		
Como as visibilidades são identificadas na maioria das linguagens de programação ?		Na grande maioria das linguagens de programação, nós atribuímos as visibilidades antes de colocar o nome dos elementos, para identificar as visibilidades usamos as seguintes palavras reservadas: - public: para representar elementos públicos; - private: para representar elementos privados; - protected: para acessar elementos que pertencem a Super Classe; <i>Lembrando que, isso não é uma regra, afinal, temos muitas linguagens que tratam o uso de classes e visibilidades de formas diferentes;</i>

[illegible]