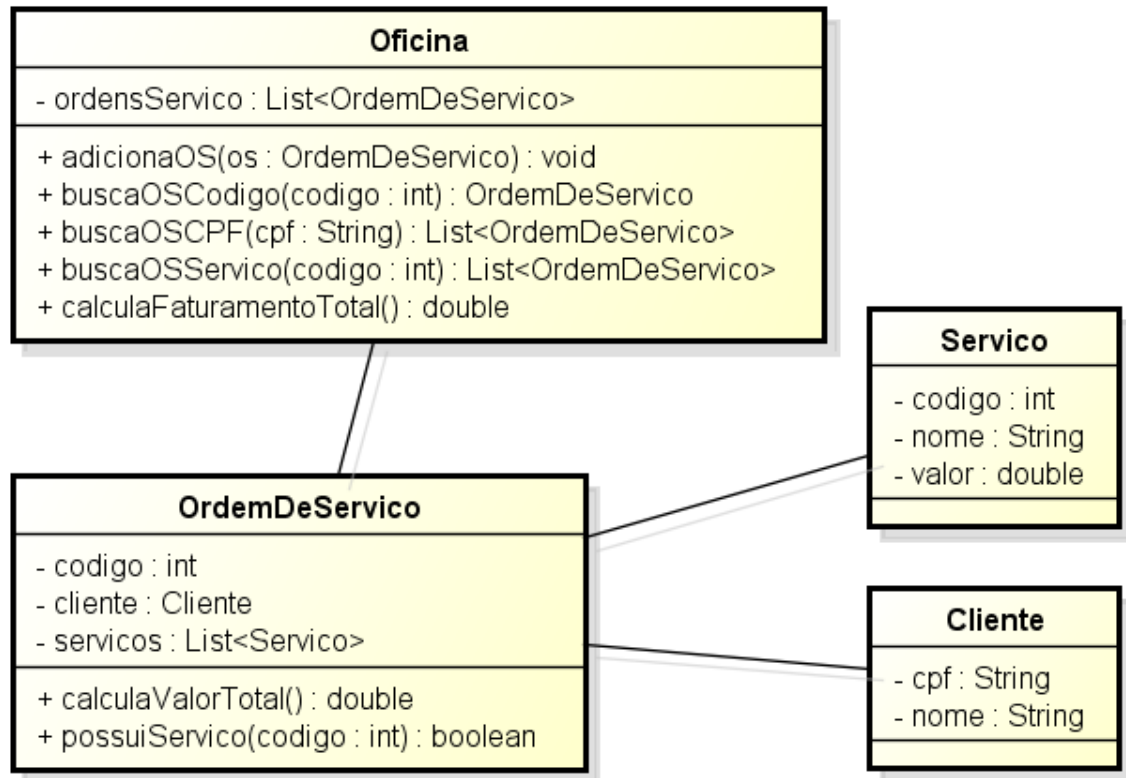




Desenvolva as classes do diagrama abaixo:



Preste atenção nos seguintes detalhes:

- Todas as classes devem possuir construtor, getters e setters e toString.
- Classe OrdemDeServico
 - o método calculaValorTotal deve somar o valor de todos os serviços e retornar a soma.
 - o método possuiServico deve receber um inteiro referente ao código de um Servico e retornar se a OrdemDeServico possui o Servico consultado.
- Classe Oficina
 - o método buscaOSCodigo deve buscar na lista de ordens de serviço a OrdemDeServico que possui determinado código ou retornar null.
 - o método buscaOSCPF deve buscar na lista de ordens de serviço as OrdemDeServico que possuem determinado cliente e retornar uma lista com essas ordens.

- o método `buscaOSServico` deve buscar na lista de ordens de serviço as `OrdemDeServico` que possuem determinado serviço e retornar uma lista com essas ordens.
 - o método `calculaFaturamentoTotal` deve somar o valor total de todas as ordens de serviço e retornar essa soma.
- Crie a classe principal a qual deve possuir o seguinte:
 - Uma `Oficina`
 - Três objetos do tipo `Servico`
 - Três objetos do tipo `Cliente`
 - Seis objetos do tipo `OrdemDeServico` de preferência com mais de um `Servico` em cada uma.
- Realize os seguintes testes:
 - Busque uma `OrdemDeServico` por código na `Oficina` e imprima o cálculo de seu valor total.
 - Busque uma lista de `OrdemDeServico` por CPF na `Oficina` e a imprima.
 - Busque uma lista de `OrdemDeServico` por `Servico` na `Oficina` e a imprima.
 - Imprima o faturamento total da oficina.