1. **Projeto Calculadora (Academia Java)**

Para esse exemplo nós temos 4 Classes, cada uma com suas responsabilidades.

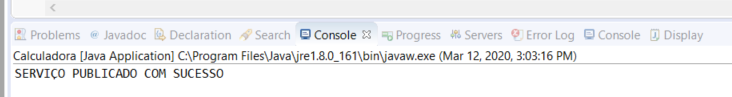
Projeto parecido com o link: <https://www.devmedia.com.br/desenvolvendo-e-usando-web-services-em-java/37261>

**Calculadora.java**

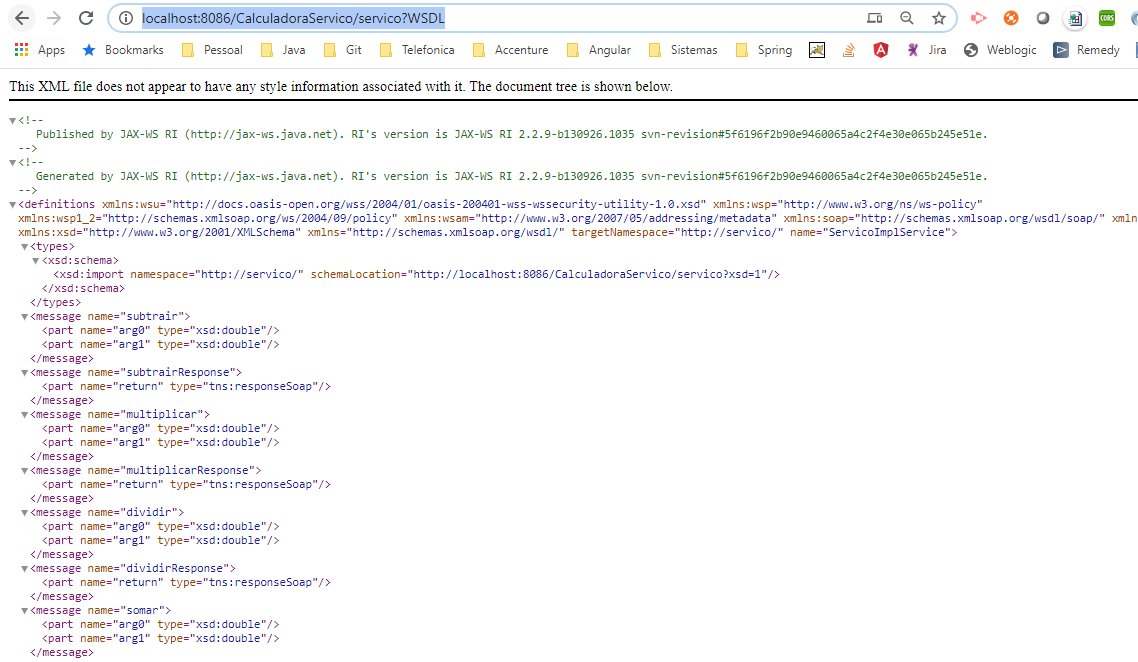
Aqui vamos publicar um enpoint na porta 8086 da máquina local.



Após executar a aplicação, nosso serviços estará exposto no endereço definido no endpoint.

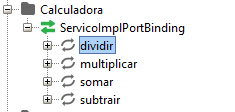


<http://localhost:8086/CalculadoraServico/servico?WSDL>



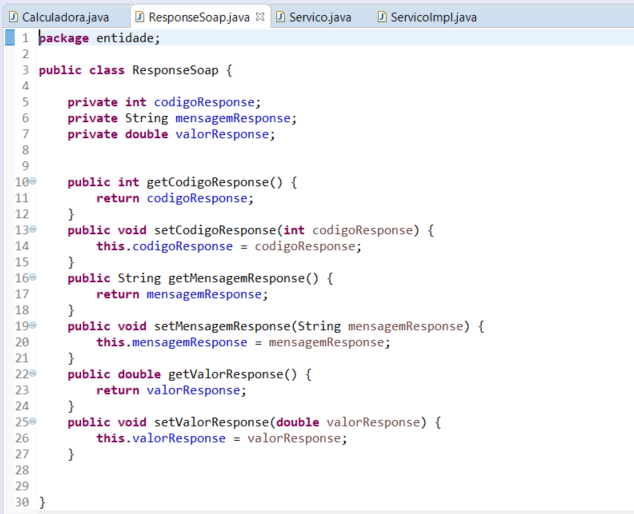
**Observação:** Lembrar de adicionar o “?WSDL” para encontrar o serviço publicado.

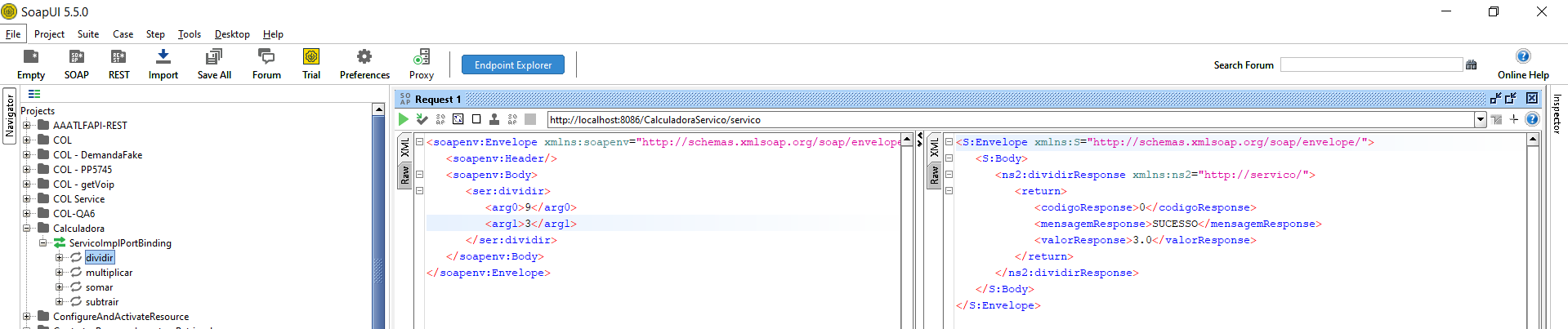
Importar WDSL no SoapUI



**ResponseSoap.java**

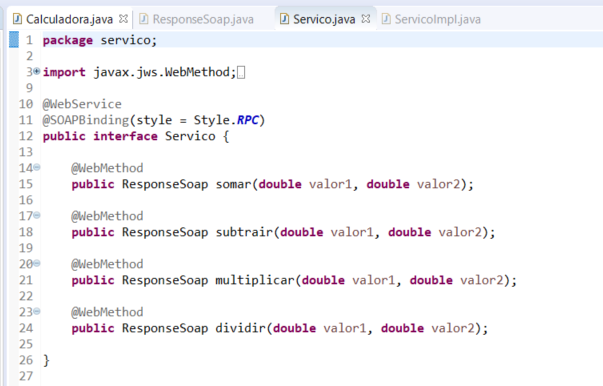
Classe que vai conter os campos da mensagem de Retorno

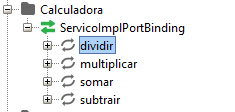




**Servico.java**

Classe que vai conter as anotações @WebService e @WebMethod. Será através dessas anotações que o serviço será criado com suas respectivas operações (somar, multiplicar, somar, dividir).



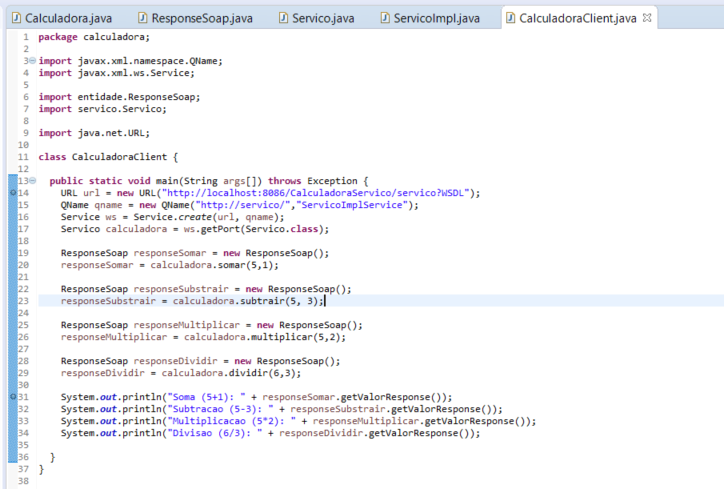


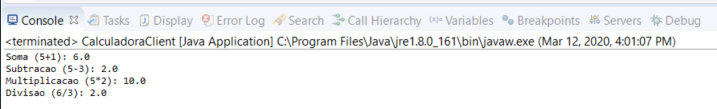
**Servico.java**



**CalculadoraClient.java**

Essa classe vai acessar o endpoint e chamar os serviços expostos e calcular os valores passados nos parâmetros.

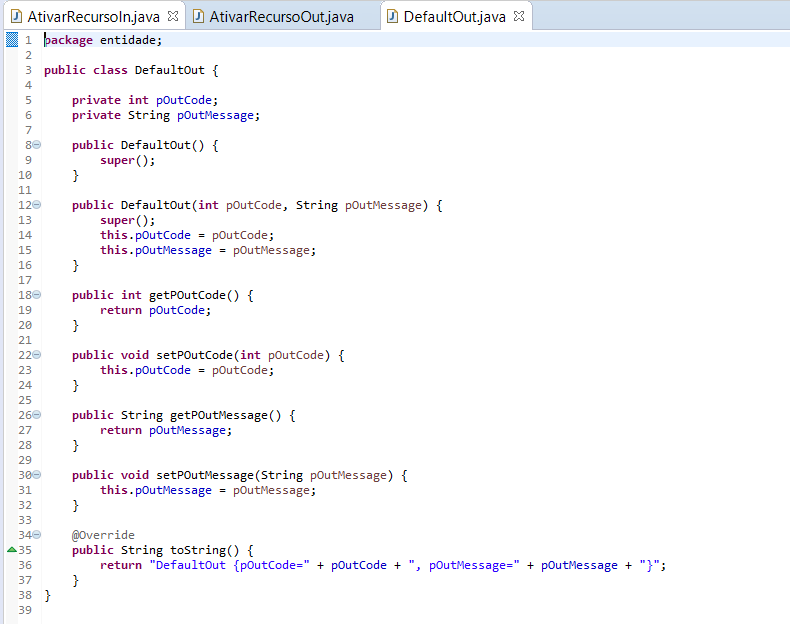




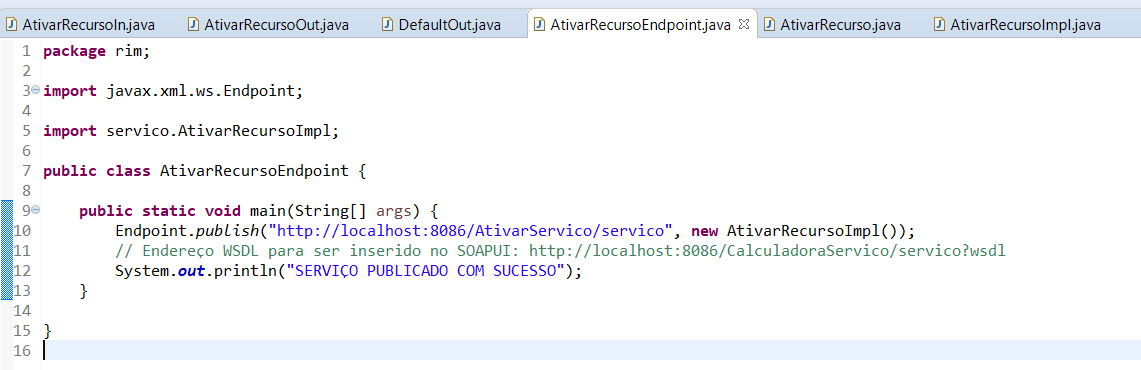
1. **Projeto AtivarRecurso (Trabalho)**

**Classes de Entidade**

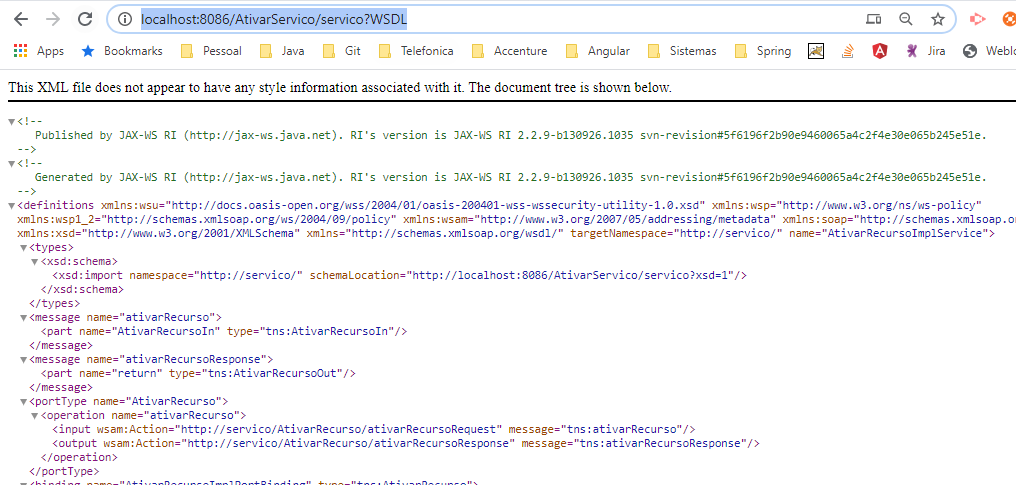
|  |  |
| --- | --- |
| **AtivarRecursoIn.java** | **AtivarRecursoOut.java** |



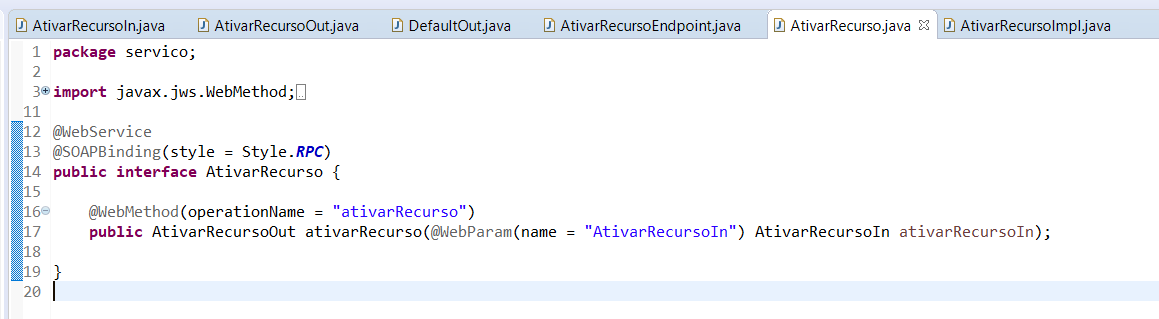
**Classes de Inicialização do Endpoint**



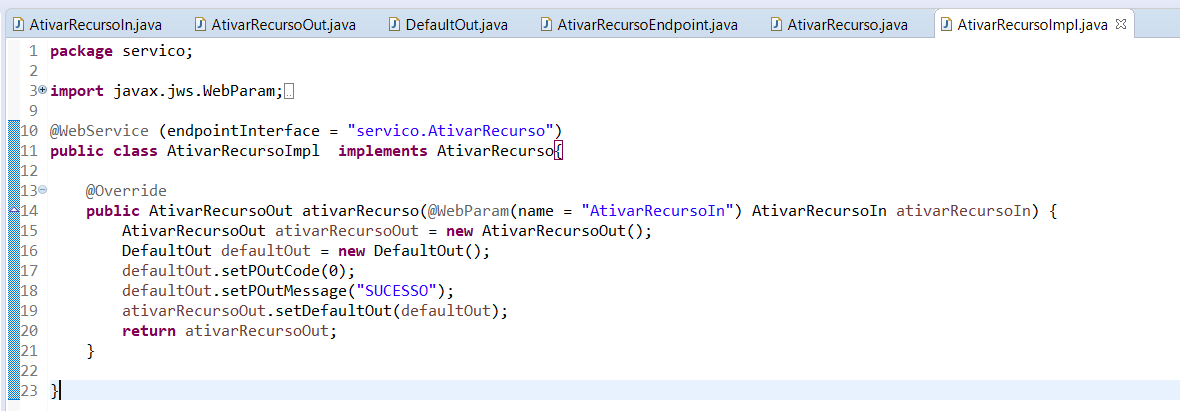
<http://localhost:8086/AtivarServico/servico?WSDL>



**Interface**



**Classe Concreta**



**Chamando o serviço**

