

**Atividades – Semana 2**

**Sistemas Web Seguros**

Atividade Formativa (on-line / pré-aula)

|  |
| --- |
| **ATIVIDADE FORMATIVA: Serviços de subtração/multiplicação/divisão e busca de UFs** |
| **ESTUDANTE: Raphael Mendes Stopa** |

**Atividade 1 – Serviço de subtração – Resolvida**

Abra seu projeto **projetosoa** e inclua, no *endpoint* CalculadoraService, o serviço de subtração. Seu serviço deve seguir o padrão de mensagem definido pelo WSDL de referência.

<message name=*"subtrair"*>

<part name=*"numero1"* type=*"xsd:int"*/>

<part name=*"numero2"* type=*"xsd:int"*/>

</message>

<message name=*"subtrairResponse"*>

<part name=*"return"* type=*"xsd:int"*/>

</message>

Entregas:

1. Trecho do código da classe Java com o método que implementa o serviço solicitado.
2. Teste o serviço com os valores -9 e 10. O resultado deve ser -19. Envie os respectivos envelopes SOAP de envio e recebimento do teste por meio de *print* de tela do SoapUI.

Resposta:

1. Trecho do código da classe Java com o método que implementa o serviço solicitado.

@WebMethod(action = "subtrair")

**public** **int** subtrair(

@WebParam(name = "numero1") **int** numero1,

@WebParam(name = "numero2") **int** numero2) {

**return** numero1 - numero2;

}

1. Respectivos envelopes SOAP de envio e recebimento do teste por meio de *print* de tela do SoapUI.

*Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente*

**Atividade 2 – Serviço de multiplicação**

Abra seu projeto **projetosoa** e inclua, no *endpoint* CalculadoraService, o serviço de multiplicação. Seu serviço deve seguir o padrão de mensagem definido pelo WSDL de referência.

<message name=*"dividir"*>

<part name=*"numero1"* type=*"xsd:double"* />

<part name=*"numero2"* type=*"xsd:double"* />

</message>

<message name=*"dividirResponse"*>

<part name=*"return"* type=*"xsd:double"* />

</message>

Entregas:

1. Trecho do código da classe Java com o método que implementa o serviço solicitado.
2. Teste o serviço com os valores 200 e -5. O serviço deve retornar -1000. Envie os respectivos envelopes SOAP de envio e recebimento do teste por meio de *print* de tela do SoapUI.

**Atividade 3 – Serviço de divisão de números decimais**

Abra seu projeto **projetosoa** e inclua, no *endpoint* CalculadoraService, o serviço de divisão. Detalhe importante: o serviço de manipulação de números decimais(*double*).Seu serviço deve seguir o padrão de mensagem definido pelo WSDL de referência.

<message name=*"dividir"*>

<part name=*"numero1"* type=*"xsd:double"* />

<part name=*"numero2"* type=*"xsd:double"* />

</message>

<message name=*"dividirResponse"*>

<part name=*"return"* type=*"xsd:double"* />

</message>

Entregas:

1. Trecho do código da classe Java com o método que implementa o serviço solicitado.
2. Teste o serviço com os valores 1 e 3. O serviço deve retornar 0.3333333333333333. Envie os respectivos envelopes SOAP de envio e recebimento do teste por meio de *print* de tela do SoapUI.

**Atividade 4 – Busca de UFs por capitais**

Abra seu projeto **projetosoa** e inclua, no *endpoint* BuscadorUFService, um serviço que busca UFs a partir do nome da capital. Por exemplo, se buscarmos pela palavra “pessoa”, retornará a UF da Paraíba, pois sua capital é João Pessoa. A busca deve ser *case-insensitive*, ou seja, buscar “Pessoa” ou “pessoa” deve resultar no mesmo retorno.

Entregas:

1. Trecho do código da classe Java com o método que implementa o serviço solicitado.
2. Teste o serviço com o valor “são”. O serviço deve retornar apenas duas UFs. Envie os respectivos envelopes SOAP de envio e recebimento do teste por meio de *print* de tela do SoapUI.

**Atividade 5 – Busca de UFs por habitantes superiores a um número**

Abra seu projeto **projetosoa** e inclua, no *endpoint* BuscadorUFService, um serviço que busca UFs cujo critério será o número de habitantes. O serviço receberá como parâmetro um número inteiro e retornará todas as UFs que possuem número de habitantes maior que o número informado no parâmetro.

Entregas:

1. Trecho do código da classe Java com o método que implementa o serviço solicitado.
2. Teste o SoapUI passando o valor “2886698” (valor da Bahia). A busca retornará apenas 3 UFs. Envie os envelopes SOAP de envio e recebimento do serviço por meio de *print* de tela do SoapUI.

**Atividade 6 – Busca de UFs por habitantes inferiores a um número**

Abra seu projeto **projetosoa** e inclua, no *endpoint* BuscadorUFService, um serviço que busca UFs cujo critério será o número de habitantes. O serviço receberá como parâmetro um número inteiro e retornará todas as UFs que possuem número de habitantes menor que o número informado no parâmetro.

Entregas:

1. Trecho do código da classe Java com o método que implementa o serviço solicitado.
2. Teste o SoapUI passando o valor “413418” (valor do Acre). A busca retornará apenas 2 UFs. Envie os envelopes SOAP de envio e recebimento do serviço por meio de *print* de tela do SoapUI.

.

